On-ground risk estimation of reentering human-made space objects

Francois Sanson

NNT : 2019SACLX035

THÈSE DE DOCTORAT

de

L'Université Paris-Saclay

École doctorale de mathématiques Hadamard (EDMH, ED 574)

Établissement d'inscription : École Polytechnique

Établissement d'accueil : Institut national de recherche en informatique et en automatique

Laboratoire d'accueil : Centre de mathématiques et de leurs applications, UMR 8536 CNRS

Spécialité de doctorat : Mathématiques Appliquées

François SANSON

Estimation du risque humain lié à la retombée d'objets spatiaux sur Terre

Thèse soutenue le 9 septembre 2019 à Palaiseau

Après avis des rapporteurs :
Thierry KLEIN (Professeur, École Nationale de l'Aviation Civile)
Rodolphe LE RICHE (Directeur de recherche, École des Mines de Saint Etienne)
Marco PANESI (Associate professor, University of Illinois at Urbana-Champaign)

Jury de soutenance :

Jean-Marc BOUILLY	(Ingénieur, ArianeGroup) Invité
Anne BOURDON	(Directrice de recherche, LPP-École Polytechnique) Présidente du jury
Pietro CONGEDO	(Chargé de recherche, CMAP-Inria SIF, École Polytechnique) Codirecteur de thèse
Thierry KLEIN	(Professeur, École Nationale de l'Aviation Civile) Rapporteur
Olivier LE MAITRE	(Directeur de recherche, CMAP- École Polytechnique) Codirecteur de thèse
Rodolphe LE RICHE	(Directeur de recherche, École des Mines de Saint Etienne) Rapporteur
Thierry MAGIN	(Associate professor, Von Karmann Institute for Fluid Dynamics) Examinateur
Marco PANESI	(Associate professor, University of Illinois at Urbana-Champaign) Rapporteur

Remerciements

Mes premiers remerciements vont aux membres du jury : Thierry Klein, Rodolphe Le Riche et Marco Panesi qui ont pris le temps de lire et de commenter ce travail avec bienveillance et attention en tant que rapporteurs. La version finale de cette thèse a grandement bénéficié de leurs remarques. Je remercie Anne Bourdon d'avoir présidé le jury ainsi que Thierry Magin en tant qu'examinateur.

Je voudrais remercier tout particulièrement mes directeurs de thèse, Pietro et Olivier, qui ont su encadrer cette thèse malgré les quelques cinq cents kilomètres qui nous séparaient. Pietro, pour avoir veillé jusqu'aux moindres détails pour que ces trois années de thèses se passent au mieux. C'est grâce à sa motivation que rien ne pourrait entamer et à son optimisme contagieux que ce travail a pu être mené à bien. Ses relectures attentives et ses nombreuses corrections ont été précieuses pendant la rédaction du manuscrit. Merci à Olivier d'avoir rejoint l'équipage en cours de route, pour les longues heures de discussion, pour ses idées et surtout pour sa rigueur que je me suis forcé à adopter autant que possible.

Ce travail s'appuie largement sur le travail d'ingénieurs passionnés à ArianeGroup : Charles Bertorello, Celia Finzi, Grégory Pinaud et François Lemaire qui m'ont apporté une aide indispensable pour comprendre les codes de réentrée atmosphérique et pour pouvoir modéliser les incertitudes. Merci pour votre patience et votre disponibilité. Je remercie en particulier Jean-Marc Bouilly pour les discussions scientifiques et pour avoir résolu un à un les obstacles administratifs/problèmes de transfert de codes/accès au CCRT etc.

Pendant cette thèse j'ai eu la chance de pouvoir collaborer avec d'autres chercheurs passionnés et pédagogues : Alessandro Turchi, Olivier Chazot et Thierry Magin lors de mes visites au VKI, Daan Crommelin au CWI et Gianluca Iaccarino à Stanford. Merci à eux.

Un grand merci à la UQ team de Bordeaux : Mickael, Andrea et Nassim pour les discussions scientifiques ou pas et le soutien face aux lots de frustrations quotidiennes que la thèse apporte. Merci aux autres membres de CARDAMOM pour leur bonne humeur et convivialité en particulier les doctorants : Andrea, Aurore, Elie, Luca et Umberto. Lors de mes passages à Saclay, j'ai aussi eu la chance de rencontrer Joao, Giulio et Nicolas. Merci à Anthony Nouy de m'avoir hébergé à Nantes pour les derniers mois de ma thèse. Enfin un immense merci à Anne-Laure et Marie, pour toutes ces missions saisies à la dernière minute et pourtant validées à temps.

Les derniers remerciements vont à ma famille : à mes parents et mes grands-parents. C'est leur soutien inconditionnel, leur bienveillance et leur patience de toujours qui m'ont amené à cette thèse. À mes frère et sœurs qui devront très respectueusement m'appeler docteur. Enfin merci à ma Lucie qui éclaire chaque jour de ma vie.

Contents

List of Figures

List of Tables

Glossary

AG ArianeGroup. 10, 35–38, 40–42, 144, 161, 170

AL Active Learning. 68, 69, 88, 129, 132, 134

ARD Atmospheric Reentry Demonstrator. 51

ASAT Anti SATellite weapon. xiii, 2

ATV Automatic Transfer Vehicle. 28, 41

CFD Computational Fluid Dynamic. 51

CFRP Carbon Fiber Reinforced Polymer. vii, 24–27, 40, 55

CMC Ceramic Matrix Composite. 24

CNES Centre National d' Etudes Spatiales. xvii, xviii, 7, 35, 36, 38, 40, 41, 55, 169, 170

DoE Design of Experiments. 67, 68

DoF Degree of Freedom. 36, 37, 43, 46, 148, 164, 170, 173, 174, 180

DRA Declared Re-entry Area. 6, 7, 31–33, 42, 50

DRAPS Debris Reentry and Ablation Prediction System. 36, 41

EGO Extended Geosynchronous Orbit. 3, 4

ESA European Space Agency. xiii, xv, xviii, 1, 6, 7, 36, 160

FSOA French Space Operation Act. 6

GEO Geosynchronous Earth Orbit. xiii, xiv, 2–4, 6, 27

GP Gaussian Process. viii, xix, 10, 11, 61, 63, 65–67, 71, 81, 127, 128, 162, 164

GTO Geo Transfer Orbit. xix, 3, 47, 101–103, 123, 137, 143, 160

GWP Global World Population. 45, 53

HTG Hyperschall Technologie Göttingen. 36

IADC Inter-Agency Space Debris Coordination Committee. 6

IGO International Governmental Organisations. 6

IMSPE Integrated Mean Square Predicted Error. 69–71, 129, 163

ISS International Space Station. xiv, 4, 28

JAXA Japan Aerospace Exploration Agency. 36

KLE Karnuhen Loeve Expansion. xi, 66, 106–108, 150, 151, 153, 154, 160

LEO Low Earth Orbit. xiii, 2–4, 6, 158

LHS Latin HyperCube Sampling. 59, 67, 69, 107, 150, 151

MC Monte Carlo. 8, 59, 68, 70, 81, 139

MCMC Monte Carlo Markov Chain. 63

MEO Medium Earth Orbit. 3

MMSPE Maximum Mean Square Predicted Error. 69–71, 129, 163

MVS Minimum Volume Sets. 33

NASA National Aeronautics and Space Administration. xv, 7, 36

ONERA Office national d'etudes et de recherches aerospatiales. 36, 51

PCE Polynomial Chaos Extension. 8, 59, 60, 67

pdf Probability Density Function. xxi

PROMES PROcedes, Materiaux et Energie Solaire. 22

QMC Quasi Monte Carlo. 59

QoI Quantity of Interest. 57–60, 105, 107, 144

RTG Radioisotope Thermo-electric Generator. 5

SA Sensitivity Analysis. 60

SoGP System of Gaussian Processes. xi, xix, xx, 10, 11, 101, 105–108, 143, 150, 151, 153, 154, 160, 162, 164, 165

SoS System of Solvers. 35, 72, 73, 150, 162

SRA Safety Re-entry Area. 6, 7, 13, 31–33, 42, 50

UP Uncertainty Propagation. 55, 59, 67, 68

UQ Uncertainty Quantification. 8, 42, 47, 60, 61, 65, 73, 161

Nomenclature

Subscripts

∞	free stream
$\mathcal{X}$	posterior prediction associated with training set $Xcal$
abl	ablation
$aero$	aerodynamic or aerothermal
$aero$	aerodynamic, aerothermal
AL	aluminum
b	survivability subspace
$breakup$	breakup
c	casualty,coriolis,cell
ch	chimistry
$cont$	continuum flow
$conv$	convective
D	drag
dyn	dynamic
e	Centrifuge acceleration
e	external,equilibrium
end	end
f	fragment
fm	free molecular flow
$frag$	fragmentation
$fusion$	fusion
geo	geodesic
h	human
il	impact location
$IMSPE$	Integrated Mean Square Predictive Error
IN	Inconel
in	input
$init$	initial
L	lift
l	rolling moment (coefficient)
M	material
m	pitching moment (coefficient)
MAP	Maximum A Posteriori
Mm	maximin
mM	minimax
$MMSPE$	Maximum Mean Square Predictive Error
n	yawing moment (coefficient)
ob	object
out	output
oxi	oxidation
p	pyrolyze
r	recovery
rad	radiative
ref	reference
res	residence
s	stagnation,static
sa	survivability assessment
sea	sea level
$shell$	object shell
$ssonic$	supersonic
ST	stainless steel
sub	subsonic
sur	survival, survivability
T	total
TI	titanium
v	validation
vib	vibrational
w	wall
Y	side force (coefficient)

Symbols

α	material absorption coefficient
α	weighting factor, deorbiting boost angle
α_E	thermal accommodation coefficient
β	function basis coefficient vector, deorbiting boost angle
δ	boundary layer thickness, normal random variable
ϵ	error, normal random variable
γ	Heat capacity ratio
G	Gaussian process
λ	mean free path, population function, logistic function, eigenvalue
$\mathbb{R}$	Real line

$\mathcal{N}$	normal distribution	H	differential entropy, hazard function,enthalpy
$\mathcal{U}$	Uniform distribution		
μ	dynamic viscosity,predictive (or posterior) mean	h	function basis vector
		h	enthalpy
ν	measure	I	Inertia matrix,mutual information, Heaviside function
Ω	Input space		
ϕ	eigenfunctions	i	orbit inclination, dummy sum index
π	constant, membership probability	K	covariance matrix
$\mathbf{\Omega}$	Object instantaneous rotation vector	k	covariance function, integer
$\mathbf{\Theta}$	Euler angles	K_n	Knudsen number
$\boldsymbol{F}$	force vector	Kn	Knudsen number
$\boldsymbol{V}$	object velocity	L	length
$\boldsymbol{X}$	object position	l	latent Gaussian Process (for the GP classifier), integer
ρ	density		
Σ	covariance function	M	Mach number, moments, model, integer
σ	Stefan-Bolztmann constant, noise, standard deviation, sigmoid	m	object mass
		P	pressure,probability
τ	time, local stress	p	locale pressure, probability density, percentage, membership probability
$\mathcal{T}$	Breakup time distribution		
ε	emissivity	Pr	Pranlt number
$\mathcal{V}$	set (unobserved location)	Q	criterion
$\mathcal{X}$	Training set, set	q	heat flux, quantity of interest
ξ	input uncertainties	R	risk
ζ	coefficients	R_n	Curvature radius
A	deorbiting boost amplitude, area, acceleration	Re	Reynolds number
		S	surface, Sobol index
a	constant in Sobol function, constant in Ishigami function, beta distribution parameter	s	local surface
		St	Staton number
		T	temperature
B	survivability assessment	t	time
C	aerodynamic coefficient	u	breakup model parameter
C	classifier, constant, computational cost	V	object velocity (amplitude), variance contribution
c	casualty factor		
c	sound velocity	v	flow velocity
D	distortion	X	random variable
d	distance	Y	random variable (output of SoGP), impact location
E	error		
F	force amplitude, kriging estimator	y	exact SoS prediction
f	function	Z	Gaussian process, normalizing constant
g	gravity acceleration		

Résumé étendu en français

Débris spatiaux, rentrée atmosphérique et risque au sol

Depuis le lancement de Sputnik-1 le 4 octobre 1957, les activités spatiales ont provoqué la mise en orbite d'un grand nombre d'objets. Si certains sont des satellites actifs et cruciaux pour les activités humaines y compris au sol, la grande majorité sont des objets non contrôlés et inutilisés, typiquement issus d'étages supérieurs de fusées, de satellites en fin de vie ou de fragments créés par érosion ou collision. Ces objets sont appelés débris spatiaux et définis par l'Agence spatiale européenne (ESA) comme "tous les objets non fonctionnels créés par l'homme, y compris leurs fragments et éléments, en orbite terrestre ou qui rentrent dans l'atmosphère terrestre" [14]. Les débris spatiaux font généralement référence à des objets envoyés par l'homme et excluent les météorites et autres types de corps célestes.

Prolifération de débris dans l'environnement spatial

Dans les premières années de la conquête spatiale, la plupart des débris spatiaux étaient des corps de fusée situés sur l'orbite terrestre basse (LEO). Le problème de la gestion des débris spatiaux s'est posé pour la première fois dans les années soixante-dix et quatre-vingt en raison d'une série d'explosions de corps de satellites et d'étages de fusée. En 1978, en particulier, l'explosion de Syncom-3 dans l'orbite geostationnaire (GEO) a donné naissance aux premières directives de réduction des débris spatiaux proposées dans [15].

Depuis les années soixante-dix, les tests anti-SATellite (ASAT) ont également contribué de manière significative à l'augmentation du nombre de fragments. La première arme antisatellite rapportée a été testée en 1968 lorsque Cosmos-249 est entré en collision volontairement avec Cosmos 248. Plus récemment, la Chine a effectué un test de missiles ASAT sur un satellite météorologique chinois FY-1D. Le test réussi a abouti à la création d'un nuage de débris de 2000 fragments de plus de 1 cm ainsi que d'un nombre de particules beaucoup plus important.

Selon le rapport annuel sur l'environnement spatial de l'ESA 2018 [16], plus de 20 000 objets catalogués gravitent autour de la Terre, mais seulement 20 % d'entre eux sont des satellites actifs. Le reste est constitué de débris spatiaux générés notamment par la fragmentation d'objets spatiaux (collision ou explosion). Ils peuvent être des corps de fusée comprenant des étages orbitaux complets de lanceurs, de moteurs ou de carénages.

L'évolution du nombre d'objets catalogués ne reflète pas exactement l'augmentation réelle des débris spatiaux, car tous les débris créés ne sont pas systématiquement détectés. Le nombre de débris spatiaux non catalogués compris entre 1 et 10 cm est estimé à environ 500 000, tandis que le nombre d'objets de moins de 1 cm dépasse probablement la dizaine de millions [17].

La rentrée atmosphérique

Le principal processus d'élimination de débris est la rentrée atmosphérique [18]. Même à des altitudes voisines de 2000 km, l'atmosphère résiduelle diminue encore la vitesse de l'objet spatial et finit par conduire à une rentrée. Le processus peut être prolongé en fonction du coefficient de traînée et de l'altitude de l'objet. Par exemple, un satellite espion à 200 km rentrera en quelques jours, et la Station spatiale internationale (ISS) devrait rentrer dans les six mois (400 km) sans reboost, tandis que le télescope Hubble reviendrait au bout de dix ans. À 825 km, la rentrée dure plus de cent ans et, dans l'orbit géostationnaire (GEO), il dépasse les millions d'années [1, 6]. En 2017, 178 événements de rentrée ont été détectés, représentant une masse de 77,5 tonnes, dont 85% proviennent de morceaux de fusées.

Une rentrée est le retour d'un objet de l'espace extra-atmosphérique où la traînée atmosphérique est négligeable vers les couches les plus denses de l'atmosphère jusqu'à l'arrivée au sol ou la destruction complète. Ce type de rentrées est dit destructrice, car on s'attend à ce que l'objet soit fragmenté et partiellement voire totalement consommé, à la fin de la rentrée, par opposition aux rentrées habitées des navettes Soyouz, par exemple. Deux types de rentrées destructrices sont possibles : la rentrée incontrôlée (ou naturelle) se produit lorsqu'un objet inutilisé quitte peu à peu son orbite d'origine en raison de la traînée atmosphérique résiduelle et sans aucune action humaine. Au contraire, la rentrée contrôlée est déclenchée par une ultime manœuvre pour désorbiter l'objet vers une région spécifique de la Terre. On note que l'objet n'est contrôlé que lors de la manœuvre initiale de désorbitation, le reste de la rentrée est totalement incontrôlé. La trajectoire de rentrée peut être décomposée en quatre grandes phases :

- La trajectoire de la haute atmosphère (120 km à 90 km) : pendant cette partie de la rentrée, l'atmosphère est environ un million de fois moins dense qu'au niveau de la mer. Les forces aérodynamiques sont limitées et l'orientation de l'objet reste stable. Le régime de vol est hypersonique dans un flux raréfié.

- La trajectoire avant la fragmentation (90 km d'altitude jusqu'à la fragmentation) : au cours de cette phase, les effets de l'atmosphère deviennent de plus en plus importants, notamment les efforts aérodynamiques et aérothermiques.

- La fragmentation : à un moment donné de la trajectoire, l'objet se fragmente. Les observations optiques de la navette spatiale Columbia, des rentrées dans l'ATV ou de la station spatiale MIR suggèrent que la fragmentation de l'objet n'est pas une fragmentation soudaine, mais une cascade de ruptures. Après la séquence de fragmentation, les fragments provenant de l'objet initial continuent de tomber sur Terre.

- La survivabilité : à ce stade, plus aucune fragmentation n'est attendue, mais chaque fragment est ralenti et ablaté en fonction de sa forme et de la composition du matériau. Certains fragments se désintègrent complètement tandis que d'autres survivent et atteignent le sol. À environ 20 km d'altitude, les fragments survivants tombent en chute libre à une vitesse subsonique en fonction de leurs propriétés aérodynamiques. En fonction des points d'impact des fragments, il est possible de calculer la probabilité de faire des victimes et leur nombre attendu.

À l'exception d'événements catastrophiques tels que la navette spatiale Columbia, la probabilité d'être percuté par un débris est extrêmement faible par rapport à d'autres sources de risque. Dans [19], le risque est estimé à 10^{-12} par an, à comparer avec la probabilité d'être frappé par la foudre (10^{-7}) ou d'avoir un accident ménager (10^{-5}) [1]. Cependant ce risque est réel et amené à croître avec l'augmentation du nombre de débris spatiaux

Risque au sol

À présent, il n'existe pas de traité international régulant la gestion des débris spatiaux et leur retombée sur Terre. Les agences spatiales telles que la NASA ou bien ESA ont formulé des règles de bonne conduite, et la loi française du 3 juin 2008 impose d'estimer le risque au sol lié au lanceur qui retombe sur Terre. Dans le cas d'une rentrée contrôlée, la zone d'impact de l'objet ne doit chevaucher les eaux territoriales d'aucun État [20]. En outre, la probabilité maximale de victimes doit être maintenue au-dessous de 2×10^{-5} [21]. Pour les rentrées non contrôlées, la probabilité de perte humaine doit être inférieure à 10^{-4} [21]. Une note technique démontrant la conformité de la mission spatiale à la réglementation française doit être présentée avant tout lancement. Elle doit inclure en particulier une estimation robuste du risque au sol lors de la rentrée.

Le calcul du risque lors de la retombée d'objets spatiaux passe par des simulations numériques qui prédisent la trajectoire de rentrée, l'effort aérodynamique, le comportement de la structure de l'objet et des échanges de chaleur. Les simulations numériques font appel à des modèles mathématiques qui approximent les phénomènes physiques. Pour une rentrée atmosphérique, ce type de calcul est complexe et couteux, notamment parce qu'il fait appel à un vaste panel de modèles physiques et solveurs associés.

Dans ce travail nous prédisons la retombé d'un objet en construisant un *système de solveurs*, c'est à dire un ensemble de solveurs où chacun résout un aspect spécifique de la physique. Nous nous intéressons en particulier aux systèmes de solveurs *dirigés*. Un système de solveurs est dit *dirigé* s'il n'y a pas de couplage fort entre les solveurs. La sortie d'un solveur peut devenir l'entrée d'un autre solveur mais la sortie de ce dernier (ou d'autres solveurs qui en dépendent) ne peut devenir l'entrée du premier solveur. Ce type de résolution à la fois modulaire et flexible permet de construire et de résoudre efficacement des problèmes multiphysiques.

La grande limitation des simulations numériques est l'inexactitude des solutions calculées. Qu'elles proviennent d'une résolution d'un modèle erroné ou bien d'une méconnaissance des conditions de rentrée, les incertitudes sont omniprésentes dans les simulations numériques de retombée d'étages. Dans ce travail nous proposons d'étudier, de modéliser et de propager les incertitudes associées à un modèle de rentrée d'objets spatiaux. Une difficulté majeure de la propagation des incertitudes est le coût de calcul. La propagation d'incertitudes induit un grand coût de calcul et d'appel aux solveurs, en particulier si le nombre d'incertitudes est élevé. En général, le coût de calcul de propagation d'incertitudes augmente exponentiellement avec le nombre de dimensions selon le principe de "fléau de la dimension".

Enjeux

L'estimation du risque repose nécessairement sur la simulation numérique des phénomènes complexes associés à la rentrée, et plus spécifiquement sur des modèles numériques multi-physiques (structure, aérodynamique, trajectoire, chimie). La physique impliquée dans ce problème est variée, complexe à modéliser, et reste parfois même incomprise. Les incertitudes sont nombreuses ; pourtant, le risque de victimes estimé doit être rapide à calculer, robuste et fiable pour assurer des opérations spatiales sûres et durables. Pour combler le fossé entre les techniques actuelles et une estimation rapide et robuste du risque au sol, les défis suivants doivent être abordés :

- **Améliorer la fiabilité des modèles numériques d'interaction fluide-structure.** La simulation numérique des phénomènes d'interaction fluide-structure et gaz-surface

le long de la trajectoire est essentielle pour prédire les forces agissant sur la surface de l'objet, le transfert d'énergie, puis fragmentation potentielle de l'objet. Du point de vue de la modélisation, ce phénomène est extrêmement riche et il n'existe que peu de données pour calibrer correctement les modèles. De plus, une simulation haute fidélité sur toute la trajectoire est très couteuse. Une approche classique consiste à utiliser des modèles simplifiés, moins couteux mais qui induisent des sources d'erreurs supplémentaires. Considérant les sources d'incertitudes déjà existantes (provenant de la physique et des modèles) et l'erreur de modèle introduite par la simplification des modèles, il est clair qu'une évaluation de la fiabilité de la prédiction numérique est primordiale pour améliorer la fiabilité de nos prédictions du risque humain.

- **Améliorer la robustesse des prédictions de fragmentation.** Le processus de fragmentation a des conséquences critiques sur l'aérodynamique de l'objet et le reste de la trajectoire, mais il s'agit également du phénomène le moins bien compris et du plus complexe à modéliser. La fragmentation est définie comme la rupture de l'objet en deux fragments ou plus et correspond aux résultats combinés d'un flux de chaleur intense, d'efforts aérodynamiques ou même de l'explosion d'un réservoir ou d'une batterie. Modéliser avec précision ces phénomènes et leurs interactions est extrêmement délicat. Les modèles de rupture les plus avancés modélisent uniquement les aspects thermiques et utilisent parfois des critères simples pour modéliser l'effort aérodynamique. Ces phénomènes sont également en concurrence : l'échauffement local de l'objet peut conduire à un affaiblissement local de la structure, alors que la charge aérodynamique peut entraîner un flambage global. Déterminer lequel se produira en premier est extrêmement difficile, mais aura également des conséquences critiques sur le reste de la rentrée.

- **Modéliser et propager les incertitudes à travers un solveur à un coût de calcul raisonnable.** Les simulateurs de rentrée d'objets spatiaux sont affectés par des incertitudes qui doivent être identifiées, modélisées et propagées *via* le simulateur pour obtenir une estimation robuste du risque. Cette estimation doit être effectuée à un coût de calcul limité, excluant ainsi les méthodes d'échantillonnage classiques. Actuellement, les incertitudes ne sont pas systématiquement prises en compte lors de la prévision de la réentrée d'objets spatiaux en raison du coût prohibitif des méthodes standard de quantification des incertitudes utilisées avec des problèmes de physique complexes tels que celui-ci.

- **Identifier les incertitudes qui affectent le plus le risque estimé au sol.** L'identification des paramètres ou des erreurs de modèle qui génèrent le plus d'incertitudes dans le modèle est cruciale pour travailler à la réduction des incertitudes dans la prévision et à l'amélioration de la fiabilité des modèles.

Contributions de la thèse

En réponse à une partie des enjeux mentionnés, nous illustrons les actions développées.

- **Construction d'un solveur numérique pour la prédiction de la rentrée**: Nous proposons deux systèmes originaux de solveurs pour prédire la rentrée contrôlée ou non d'objets spatiaux. Comparés au logiciel de rentrée existant, les modèles construits incluent naturellement l'incertitude dans les prévisions de fragmentation. De plus, les solveurs sont interfacés et couplés dans un cadre unifié pour permettre des lancements automatiques et parallèles sans action de la part de l'utilisateur. Les deux simulateurs présentent différents niveaux de fidélité.

- **Proposition de modèles de fragmentation sous incertitudes**: Nous examinons les modèles disponibles pour prédire la réentrée d'un objet spatial, nous abordons les sources d'incertitude et les erreurs dans les modèles. Sur la base de cette discussion, nous développons deux modèles de fragmentation tenant compte des incertitudes de la modélisation de la rupture.

- **Développement d'outils efficaces de propagation des incertitudes pour les systèmes de solveurs** : Nous présentons deux solutions originales pour propager les incertitudes dans le système de solveurs et les prédicteurs de rentrée. i) Premièrement, on introduit une méthode pour construire efficacement des modèles de substitution pour des systèmes de solveurs dirigés. Le cadre comprend également des stratégies d'apprentissage actif efficaces pour améliorer la performance du modèle de substitution. Le cadre développé est général et peut s'appliquer à un grand nombre de problèmes industriels. ii) Deuxièmement, on présente une nouvelle formulation de l'évaluation de la survie des fragments qui permet de construire des modèles de substitution et d'effectuer un apprentissage actif.

- **Caractérisation des sources d'incertitudes dans le système de solveur** : Nous étudions les modèles disponibles pour prédire la rentrée d'un objet, et nous identifions les sources d'incertitudes et les erreurs dans les modèles. Nous proposons ensuite de les modéliser en tant que variables aléatoires et d'étudier leur influence sur un problème de rentrée d'un étage supérieur. Nous effectuons une analyse de la sensibilité de la sortie du modèle aux entrées incertaines. Nous sommes en mesure de classer les dizaines d'incertitudes en fonction de leur influence sur les incertitudes des quantités d'intérêts (par exemple le risque au sol) et de formuler des recommandations pour réduire ces dernières.

- **Preuve de concept des outils de quantification d'incertitude développés avec application au simulateur de rentrée moyenne fidélité pour prédire la rentrée d'un étage supérieur sous incertitude.** Nous développons un outil logiciel de prédiction de rentrée sous incertitude, en combinant les différentes contributions de ce travail : les simulateurs de rentrée avec le modèle de rupture probabiliste et les modèles de substitution. L'outil logiciel est utilisé pour effectuer une analyse physique des incertitudes sur la base d'une analyse de sensibilité rigoureuse et d'une estimation robuste de la quantité d'intérêt (conditions de vol, de rupture, risque au sol). Cette analyse est effectuée à un coût de calcul très raisonnable.

Les résultats principaux obtenus sont détaillés dans les paragraphes suivants.

Construction d'un simulateur de rentrée et identification des incertitudes

Nous construisons un système de solveur pour prédire la retombée d'un objet spatial. L'objectif est de construire un outil qui prédise la rentrée complète et qui puisse fournir une estimation du risque au sol. Il doit être simple d'utilisation, robuste et facilement automatisable pour être couplé avec des méthodes de quantification d'incertitudes.

Le simulateur construit est composé de cinq à six solveurs développés par le CNES et ArianeGroup et fonctionne ainsi :

1. À 120 km d'altitude, les conditions de vol de l'objet (vitesse, position) sont soit données par l'utilisateur (rentrée contrôlée), soit calculées par le solveur de désorbitation.

2. Les coefficients aérodynamiques associés à la géométrie de l'objet sont calculés par un solveur aérodynamique (ARPEGE). Afin d'éviter d'appeler le solveur aérodynamique à chaque pas de temps de la trajectoire, des tables aérodynamiques sont générées. Les tables sont fonction de l'altitude de l'objet (orientation), du nombre de Mach et du nombre de Knudsen.

3. Dans l'étape suivante, le solveur de trajectoire BL43 avec 3 degrés de liberté est couplé à un module thermique (hérité d'ADRYANS V4) et au modèle d'atmosphère MSIS-00 pour calculer la trajectoire de l'objet (avant toute fragmentation). La trajectoire inclut la vitesse, la position, la masse et la température de l'objet, qui sont ensuite nécessaires pour calculer les conditions de fragmentation.

4. L'outil de fragmentation utilise l'évolution de la température et la trajectoire pour calculer les conditions de vol lors de la fragmentation et la libération des fragments. En ce qui concerne les modèles orientés objet, la forme et les caractéristiques des fragments lors de la rupture sont précalculées par des experts à l' aide de simulations numériques sur des modèles avancés.

5. Étant données les conditions de libération de chaque fragment (vitesse, position, température) et ses caractéristiques (composition du matériau, forme, dimension, mouvement d'attitude), sa trajectoire, sa température et sa masse sont calculées. Deux cas peuvent être observés. Soit le fragment disparaît dans l'atmosphère, soit il survit et s'écrase au sol. Dans ce dernier cas, le solveur renvoie l'emplacement de l'impact (latitude et longitude), la masse résiduelle, la vitesse et l'aire meurtrie.

6. Pour les fragments survivants, le solveur d'estimation du risque calcule le risque humain.

Le couplage de codes écrits dans des langages de programmation différents a requis l' implémentation d'un cadre général écrit en python pour permettre l'interfaçage et l'appel en parallèle des solveurs.

Dans la littérature, il existe deux classes principales de prédicteur de rentrée. Les modèles orientés objet et les modèles orientés engin. La fragmentation est modélisée de manière simpliste en supposant une altitude fixe de fragmentation. La fragmentation de l'objet suit une approche parent-enfant : à la fragmentation, l'objet parent disparait et relâche les objets enfants. Les outils orientés engin utilisent une modélisation de l'objet plus détaillée et des modèles aérodynamiques plus avancés. La fragmentation est aussi directement simulée en étudiant la rupture de la structure de l'engin au cours de la trajectoire.

Comparé à des codes orientés objets tels que DEBRISK (CNES) ou DRAMA (ESA), notre modèle de fragmentation est capable de prendre en compte les incertitudes engendrées par l'absence de modélisation fine de la structure de l'objet et de sa fragmentation. Comparé à des approches orientées engins, notre modèle prends une approche plus simple dans la modélisation thermique (à la fois pour la réponse thermique de l'objet et la modélisation des échanges thermiques). Néanmoins il peut être plus facilement couplé à des approches de propagations des incertitudes.

Dans ce chapitre, nous identifions aussi les sources d'incertitudes majeures à travers deux classes d'incertitudes : les erreurs de modèles et les méconnaissances du scénario. Les erreurs de modèles proviennent notamment des modèles aérodynamiques simplifés tels que la loi de Newton modifiée, ou bien de corrélations pour les modèles thermiques, ou des interactions gaz-surface non modélisées. Les incertitudes dues à la méconnaissance de la situation sont par exemple les

conditions initiales, les propriétés de l'atmosphère, les caractéristiques des matériaux ou bien une connaissance partielle de la géométrie de l'objet.

Développement d'une méthode de propagation des incertitudes dans des systèmes de solveurs dirigés

Nous avons développé des méthodes basées sur des systèmes de processus gaussiens SoGP comme modèle de substitution de systèmes de solveurs dirigés. Dans notre approche, chaque solveur du système est approximé par un processus gaussien (GP). Le SoGP résulte de la composition de ces GP.

Nous avons analysé aussi plusieurs stratégies de générations de plan d'entrainement, certaines étant *a priori* (sans connaissance préalable du système) alors que d'autres reposent sur des méthodes d'apprentissage actif. Les méthodes d'apprentissage actif développées tentent de construire des plans d'entrainement adaptés à la structure du système de solveurs. Elles se fondent sur la distribution prédictive générée par le SoGP. Ces approches dépendent d'une décomposition de la variance prédictive de la sortie du SoGP en contribution des GPs. Cette décomposition, développée pour ce travail, permet d'identifier les GP qui sont les moins fiables dans la prédiction globale et de les améliorer de manière ciblée. Nous déclinons trois méthodes d'apprentissage actif pour SoGP utilisant le principe de variance prédictive maximale et la décomposition de variance. La méthode la plus parcimonieuse propose des points d'entrainement au GP le moins fiable et à l'endroit où il est le moins fiable (c.a.d. au maximum de contribution à la variance du SoGP).

Les performances du SoGP sont comparées à une approximation de l'ensemble du système de processus gaussien par un unique GP. Pour tous les tests, à la fois sur des fonctions analytiques ou des cas physiques, le SoGP présente des résultats bien meilleurs. Ces bons résultats de la part du SoGP par rapport à une méthode de GP unique viennent notamment de son expressivité accrue conférée par la composition de plusieurs GP, mais aussi de sa capacité à inclure, dans son plan d'entrainement les grandeurs intermédiaires.

De plus ces résultats sont encore significativement améliorés lorsque les méthodes d'apprentissage actif sont employées, en particulier si la dimension des solveurs est moyenne ou réduite. Pour des cas à hautes dimensions ($\simeq 15$) la performance des méthodes d'apprentissage actif se détériore, notamment à cause du critère d'enrichissement fondé sur le principe de la variance prédictive maximale. Il est à noter cependant que nos tests n'ont pas observé une perte en performance de l'approche SoGPen elle-même, même en haute dimension, mais seulement des stratégies d'enrichissement.

Prédiction sous incertitudes de la fragmentation d'un étage supérieur

Mise en place du problème

Nous utilisons le cadre SoGP couplé avec le modèle de rentrée proposé pour établir des prévisions fiables concernant la rentrée d'un étage supérieur depuis une orbite de transfert vers l'orbite Géostationnaire (GTO). La rentrée de l'objet est contrôlée, c'est-à-dire que l'objet exécute une manœuvre de désorbitation pour quitter la GTO et rentre dans l'atmosphère dans une région ciblée. Ce cas présente un intérêt particulier pour ArianeGroup et est donc étudié en profondeur. L'objectif est de fournir une estimation robuste des conditions de fragmentation, d'identifier les contributions de chaque incertitude aux conditions de fragmentation et d'étudier les distributions des grandeurs clés telles que la température de l'objet ou le flux thermique. Les incertitudes

d'entrée sont les éléments orbitaux de l'orbite initiale, les paramètres de désorbitation, les paramètres du modèle d'atmosphère (activité solaire, instant de rentrée, activité géomagnétique), les propriétés matériaux de l'objet et deux paramètres issus du modèle de fragmentation.

Construction du modèle de substitution pour la propagation d'incertitudes

Les incertitudes sont ensuite propagées pour reconstruire plusieurs quantités d'intérêt telles que les conditions de vol de l'objet lors de la fragmentation, la température de l'objet et le flux de chaleur convectif. Pour ce faire, un SoGP est construit et validé en le comparant à une approximation par un GP unique. Dans ce cas haute dimension (28 incertitudes), le SoGP présente une erreur d'approximation 5 à 10 fois meilleure. Cette étude confirme l'intérêt du cadre SoGP pour mener des études de propagation d'incertitude à moindre coût.

Resultats

Grâce au SoGP, une étude de propagation d'incertitude est réalisée. Nous étudions les conditions de vol de l'objet lorsque celui-ci se fragmente (position et vitesse) ainsi que l'évolution des incertitudes sur la position, vitesse, température et flux thermique le long de la trajectoire de l'objet. On constate notamment une forte augmentation de celles-ci entre 90 km et la fragmentation.

Le SoGP permet aussi de réaliser une analyse de sensibilité fondée sur la décomposition de variance des quantités d'intérêt qui aurait autrement été extrêmement coûteuse à réaliser. Cette analyse de sensibilité, effectuée sur les conditions de vol au moment de la fragmentation et sur les caractéristiques thermiques de l'objet le long de la trajectoire, permet de bien identifier l'influence de chacune des vingt-huit incertitudes sur la précision des prédictions. On observe par exemple que les incertitudes liées à l'atmosphère influencent principalement la trajectoire de l'objet à haute altitude et deviennent ensuite négligeables. De manière générale, les incertitudes les plus influentes sont les conditions de désorbitations et les paramètres du modèle de fragmentation. En particulier, l'incertitude générée par le modèle de fragmentation représente plus de 80 % de la variance de la vitesse de l'objet lors de la fragmentation. Cette analyse de sensibilité est précieuse pour orienter les futurs développements du modèle de rentrée, en identifiant les paramètres et modèles les plus influents et en permettant de négliger les incertitudes sur les paramètres non influents.

Construction d'un modèle de substitution pour prédire la survie d'un objet spatial

Formulation du problème

Nous visons à compléter l'analyse des incertitudes et à construire un modèle de substitution de l'étape de survie des fragments générés par la fragmentation de l'objet initial. Après la fragmentation de l'objet, les fragments sont libérés et s'exposent au flux externe en retombant sur Terre. Dans certains cas, les fragments peuvent être entièrement ablatés et disparaitre dans l'atmosphère en fonction de la composition du matériau et de la quantité d'énergie qu'ils reçoivent de l'écoulement.

Dans d'autres cas, le fragment survit à la rentrée et s'écrase sur Terre. Dans le modèle de rentrée proposé, le solveur de calcul de survie évalue si le fragment atteint le sol et, le cas échéant calcule l'emplacement de l'impact. Si le fragment brule dans l'atmosphère, le nombre moyen de victime associé est nul. S'il atteint le sol, le module de calcul de risque évalue le nombre moyen

de victimes en fonction de la taille de l'objet et de la densité de population au point d'impact. *A priori*, le nombre moyen de victime est une fonction discontinue des incertitudes et ainsi la construction d'un modèle de substitution peut s'avérer complexe.

Nous proposons également une nouvelle approche de modèle de substitution pour approximer le solveur de survie et calculer le nombre de victimes moyen. Nous assimilons la correspondance entre le nombre de victimes et les incertitudes d'entrée, à la composition d'un classificateur qui évalue la capacité de survie d'un fragment, et d'une fonction continue qui relie les incertitudes d'entrée aux emplacements de l'impact si l'objet survit. Cela permet de construire un modèle de substitution pour prévoir le nombre moyen de victimes et d'approximer efficacement la discontinuité du risque.

Modèle de substitution utilisé

Cette formulation permet de construire un modèle de substitution à l'aide de GP utilisés comme un classifieur et d'une régression par processus gaussiens. L'utilisation de GP permet de calculer la variance prédictive du modèle de substitution et d'en déduire des stratégies d'apprentissage actif visant à réduire l'erreur sur le risque prédit. Notamment, nous montrons que la variance de prédiction peut être décomposée en contributions de la part du classifieur ou bien de la régression. Nous proposons une stratégie d'apprentissage actif fondée sur la variance prédictive du modèle. Cette stratégie a l'originalité d'être intrinsèquement aléatoire. Au lieu de proposer des points là où la variance de prédictions est maximale (ce qui est très inefficace dans le cas de classifieur), ou bien de se baser sur une estimation de la réduction espérée de la variance, le plan d'entrainement est enrichi par des points tirés selon une distribution dont la pdf est proportionnelle à la variance prédictive. Cette approche permet à la fois de d'améliorer les prédictions dans les régions où la prédiction est incertaine sans se focaliser exclusivement sur les maxima qui sont situés le long de la frontière de classification.

Cette approche est testée sur un cas analytique en faisant varier la distribution de population et donc le risque au sol. Dans tous les cas, les méthodes d'apprentissage actifs offrent une amélioration significative par rapport à un plan d'entrainement généré aléatoirement. Ce constat est confirmé avec un test réaliste de la rétombée d'un débris sur des îles imaginaires dans le pacifique.

Application à la prédiction de la retombée controlée d'un étage superieur et calcul du risque

Mise en place du problème

Les modèles de substitutions développés auparavant sont couplés avec le simulateur de rentrée pour quantifier les incertitudes lors de la retombée sur Terre d'un étage supérieur de fusée. On complète l'étude ici avec la prédiction de la survie des fragments générés après la fragmentation et leur point d'impact ainsi que le risque humain associé.

Le modèle numérique comprend la désorbitation de l'objet, le calcul de trajectoire jusqu'à sa fragmentation, le calcul des conditions de fragmentations avec le modèle probabiliste, le calcul de la survie des fragments, la détermination des points d'impact des fragments survivants et enfin le calcul du risque au sol.

En plus du caractère aléatoire du modèle de fragmentation, un nombre conséquent (38) d'incertitudes sont considérées. On inclut notamment les incertitudes sur l'orbite initiale, les

conditions de désorbitation, les caractéristiques du matériau de l'étage, les paramètres du modèle de fragmentation, les entrées du modèle atmosphérique et les caractéristiques des matériaux constituants les fragments. Afin de propager ces incertitudes efficacement nous couplons le modèle de substitution utilisé pour prédire les conditions de fragmentation avec le modèle de substitution conçu pour prédire la survie des débris issus de la fragmentation.

Construction des modèles de substitution

Nous construisons un modèle de substitution afin de pouvoir approximer l'ensemble du simulateur de rentrée. Sachant que nous avons déjà construit un modèle de substitution jusqu'à la fragmentation, il suffit d'en construire un supplémentaire pour prédire la survie de chacun des fragments ainsi le risque associé

La construction des modèles de substitution nécessite quelques précautions par rapport aux méthodes développées précédemment pour prédire la survie des fragments. Pour certains fragments, il n'a pas pu être observé de situation où l'objet atteignait le sol parmi les points du plan d'entrainement. Bien que cela ne prouve pas que le fragment n'atteigne jamais le sol, nous négligeons ces fragments dans notre analyse du risque. Des tests de validations s'appuyant sur un très grand nombre d'échantillons confirment que l'erreur induite est acceptable.

Pour les fragments qui ont une probabilité non nulle d'arriver au sol, un modèle de substitution du solveur de survie, construit avec un classifieur et une régression par GP a été appris. Il convient de noter qu'un modèle de substitution est appris pour chaque fragment.

De plus, puisque l'objet retombe dans l'océan Pacifique, le risque de perte humaine est nul (selon les modèles de population). Ainsi la mise en place de stratégies d'apprentissage actif permettant d'améliorer la prédiction du risque n'ont pas d'intérêt dans ce cas particulier. À la place nous adaptons la méthode pour améliorer la prédiction de l'aire meurtrie (casualty area) qui correspond à la surface projetée du débris augmentée de la surface d'un humain debout. L'aire meurtrie quantifie la dangerosité d'un objet indépendamment de la densité de population. Les procédures d'apprentissage actif permettent de gagner une précision significative pour l'ensemble des débris par rapport au plan d'expérience initial.

La précision des modèles de substitution est quantifiée sur un échantillon de validation. L'erreur sur les prédictions de survie est de l'ordre de 10^{-3} pour l'ensemble des débris tandis que l'erreur de prédiction sur la position d'impact peut atteindre un kilomètre sur la surface de la Terre. Cette erreur est à comparer avec les 260 000 km^2 couverts par les distributions d'impact.

Résultats

À l'aide des modèles de substitution, nous calculons : les probabilités de survie pour chaque fragment, les distributions d'impact en cas de survie et la distribution de la surface meurtrie totale. Nous effectuons aussi une analyse de sensibilité pour mesurer l'influence des incertitudes d'entrée sur l'incertitude sur la surface meurtrie. Nous montrons que les incertitudes dominantes sont liées aux paramètres du modèles de fragmentation, aux conditions de désorbitation et dans une moindre mesure aux incertitudes sur les caractéristiques des matériaux, notamment l'émissivité. Cette étude réalisée sur un cas réel de quantification du risque sous incertitudes lors de la retombée d'un étage supérieur démontre l'intérêt de nos méthodes de substitution de modèle pour propager les incertitudes à un coût de calcul raisonnable.

Chapter 1

The Space Debris Issue

In this chapter, we discuss the impact of the space debris population on future space activities and human ground assets. We briefly present the policies established by space agencies and states to regulate the space debris proliferation phenomenon and discuss how these regulations rely on numerical simulation for the predicting the on-ground human risk during a space object reentry. In this chapter, we illustrate the significant challenges that should be tackled to robustly estimate the on-ground risk and describe the main objectives and contributions of this thesis.

1.1 Historical considerations on space debris

Since the launch of Sputnik-1 on October 4^{th} 1957, human space activities have generated a large number of objects that are currently orbiting around the Earth. While some are active satellites crucial for man on-ground activities, the vast majority are inertial objects typically coming from defunct rocket upper stages, end-of-life satellites or fragments created by erosion or collision. Those objects are called space debris and defined by the European Space Agency (ESA) as "all non-functional, human-made objects, including fragments and elements thereof, in Earth orbit or re-entering into Earth's atmosphere" [14]. Space debris commonly refers to human-made objects and excludes meteorites and other types of celestial bodies.

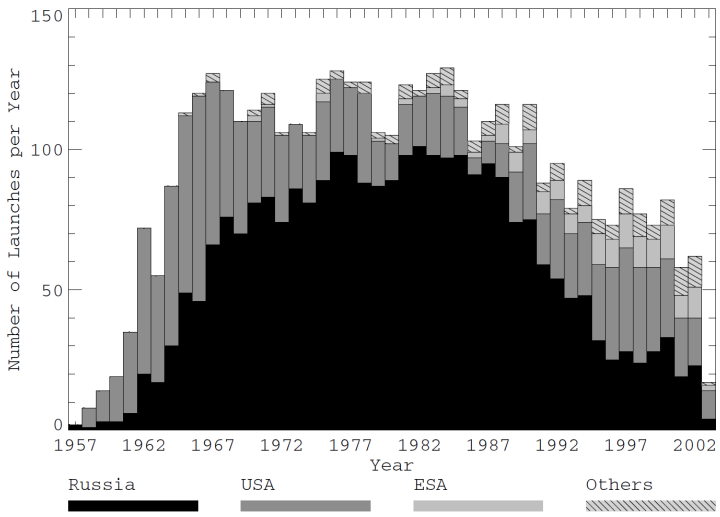


Figure 1.1: Evolution of the number of launches per launch year (source : [1])

When Sputnik-1, the first human-made satellite, was launched, the Semiorka rocket stage and a protective fairing were also on the same orbit. In total, Sputnik only represented 1.3 % of the overall mass, around 6,700 kg. In the following years, most of the space debris population was composed of rocket bodies located on the Low Earth Orbit (LEO). The problem of space debris management arose for the first time in the seventies and the eighties due to a sequence of satellite and rocket body explosions. In particular, in 1978, the blast of Syncom-3 in GEO led to the first space debris mitigation guidelines proposed in [15].

In the same years, Kessler and Cour-Palais theorized the formation of a debris belt around the Earth [18]. This debris belt could be self-sustained or even grow due to in-orbit collisions, even with no further launches [22]. The first accidental known collision occurred between the satellite Cerise and a fragment of an Ariane 1 rocket body [1] in 1996. As predicted by Kessler, each collision creates a vast amount of debris. A recent example of collision occurred in 2009 between Iridium 33 (USA) and Cosmos-2251 at a relative velocity of 11km/s resulting in the creation of thousands of debris [23].
Since the seventies, Anti-SATellite (ASAT) tests have also been a significant contributor to the increase in the number of fragments. The first reported anti-satellite weapon was tested in 1968 when Cosmos-249 collided on purpose into Cosmos 248. Other ASAT tests followed with for instance the destruction of the Solwind P78-1 destroyed with an ASAT missile. More recently China conducted a ASAT missile test on a Chinese weather satellite FY-1D. The successful test resulted in the creation of a debris cloud of 2000 fragments larger than 1cm and possibly a much larger number of particles.

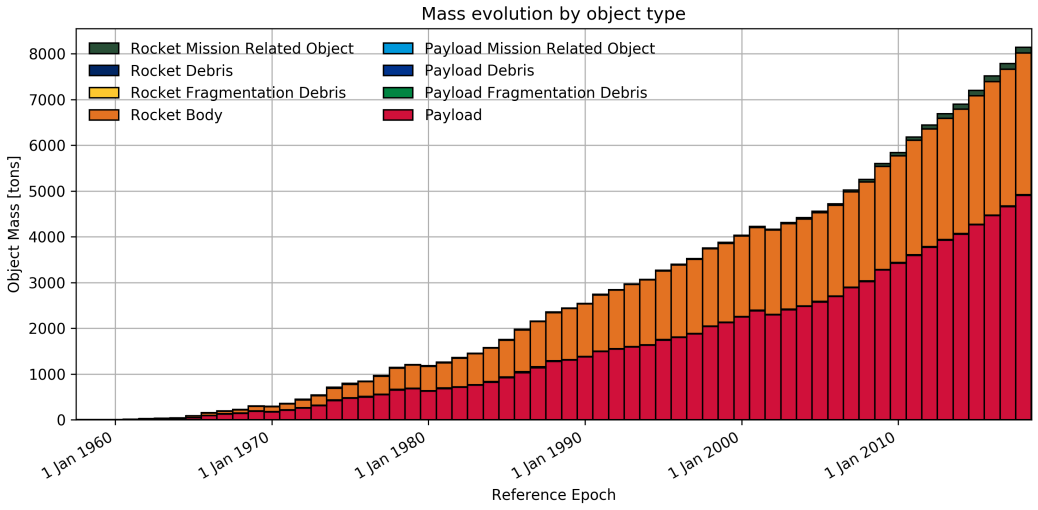


Figure 1.2: Evolution of the mass of catalogued objects by object type source: ESA DISCO tool [2]

In Fig. 1.1, the number of launches per year since 1957 are represented. Within the first ten years, the number of space launches reached a plateau from the early seventies to the early nineties. The number of launches then decreased mostly due to the collapse of the USSR in 1991. In Figs. 1.3 and 1.2, we observe that the number of cataloged objects and their mass increased linearly before early 2000, but the trend is now quadratic or faster. This trend is even more noticeable when considering Fig. 1.3 that shows the evolution of the number of space debris since 1960. Here the trend seems to be linear for the number of payload objects or rocket body, but the number of fragments follows a much faster increase, mostly due to sequences of

catastrophic collisions and satellite destruction between 2000-2010. Note that part of the rise in the number of cataloged objects can also be explained by the improved tracking capabilities of the community. The total number of fragmentation debris (rocket fragmentation debris and payload fragmentation debris) is now much higher than the number of payload objects in orbit.

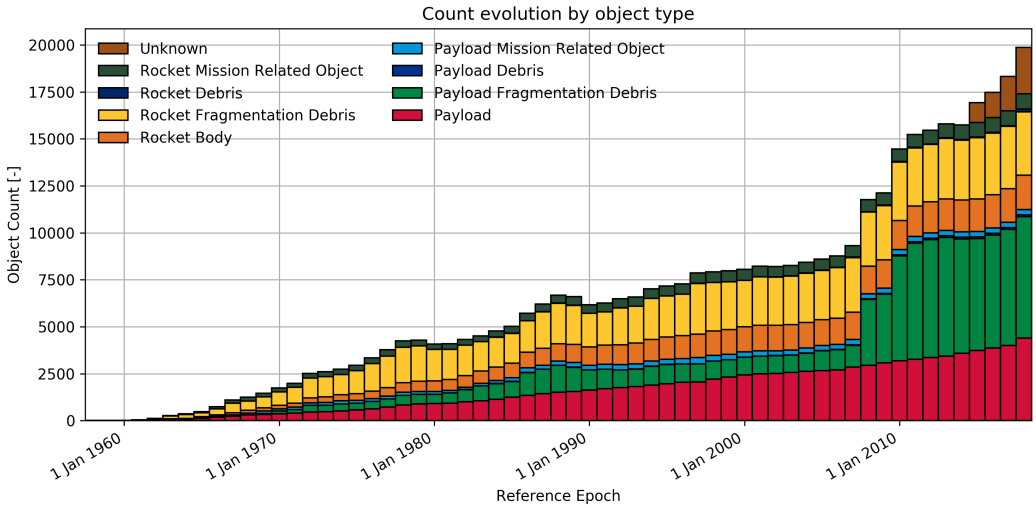


Figure 1.3: Evolution of the number of catalogued objects by object type (source: ESA DISCO tool [2])

1.2 Space debris issue

In this section, we depict the current and future state of the space debris population and the consequences for space activities and human populations.

1.2.1 Current situation

According to ESA 2018 annual space environment report [16], over 20 000 cataloged objects are orbiting around the Earth, but only 20 % are payload and active satellites. The rest is space debris. The debris can be generated by operating space objects (astronaut tools or optical instrument covers) or after a fragmentation (collision or explosion). Around 40 % of cataloged debris is associated with payloads or former satellites while 27 % is rocket bodies or derived from rocket bodies.

The majority of the objects is in LEO, that is at an altitude below 2000km, and only 10 % are in Medium Earth Orbit (MEO) or Extended Geosynchronous Orbit (EGO). The most populated orbits are the Geosynchronous Transfer Orbit (GTO) and LEO and to a lesser extent the GEO [16]. Even within the LEO, the space debris typically originates from end-of-life satellites and is concentrated around 1500 km and 800 km while debris in GTO is mostly rocket bodies.

In 2017, the total number of additional cataloged objects increased by 10 % corresponding to a newly added mass of 558 tons [16], among which 60 % were operating satellites. Hence, most of the newly added mass comes from operating objects, but in term of the number of objects, the operating objects represent only 25 % of the total number of newly cataloged objects while the rest is space debris. More than half of the newly added space objects are in LEO orbit. The number of objects is on the rise mostly due to the proliferation of small payloads (below 10kg)

that have a limited impact on the mass increase. In 2017, 4 breakups have been detected in GEO and EGO inducing an additional 23 catalogued objects.

The number of cataloged objects does not directly reflect the actual increase in space debris first because existing debris can be detected and because newly generated debris may remain undetected. The number of uncatalogued space debris between 1 and 10 cm is extrapolated around 500 000 while the number of objects smaller than 1cm probably exceeds tens of millions [17].

The main debris "sink" is space object reentry [18]. Even at altitudes around 2000 km, the atmosphere still reduces the space object velocity and eventually leads to a reentry. The process can be prolonged depending on the object drag coefficient and altitude. For instance, a spy satellite at 200 km reenters in a matter of days, the International Space Station (ISS) is expected to reenter within six months (400 km) without reboosting while the Hubble telescope would return after ten years. At 825 km, the reentry time is over a hundred years and in GEO it is over a million years [1, 6]. In 2017, 178 reentry events were detected, representing a reentering mass of 77.5 tons. Around 85 % of this mass came from rocket bodies.

1.2.2 Future evolution of the space debris population

Making a quantitative prediction of the space debris population evolution is extraordinarily challenging and subject to uncertainties [24, 25] but studies agree that in a business-as-usual scenario, the number of debris in LEO will proliferate exponentially mostly due to collision cascade [18, 1, 26, 27, 28, 29]. This trend may be alleviated if the future space missions comply with international guidelines [30]. The number of space objects in LEO may also be affected by the new actors in the space industry. The cost of rocket launches reduced significantly, and satellites are more and more used for new applications. In general, cheaper and smaller satellites are expected to be launched in LEO orbit in the future. Projects such as OneWeb, Starlink from SpaceX require mega constellations of satellites. As a consequence, the number of space debris is expected to increase by 100 % in the LEO region, between 700-1200 km [31].

The increase in the number of satellites also induces a change of paradigm for satellite manufacturers that need to mass produce satellites cheaper with a larger failure tolerance. In such cases, constellations may worsen the space debris problem, especially in the upper LEO. In [32], the authors show that a high disposal rate of end-of-life satellites is vital to maintaining a sustainable space environment. In the future, if the launch of massive constellations is confirmed, strict compliance with the existing guidelines will become a necessary condition to maintain a safe and accessible near-Earth space environment.

1.2.3 Consequences for near-Earth space activities

The sharp increase in the number of space debris especially in LEO brings additional risk and constraints for current and future space missions [33]. The collision probability is directly related to the amount of space debris in a given orbit. We can identify two types of in-orbit collisions: micro collisions or catastrophic collisions.

Micro collisions are collisions between micro debris (1 mm) and large debris and depend on the impact location, incidence, and relative velocity. Solar panels are the most exposed parts of a space object. The micro collision risk is mitigated with protective layers as for pressurized modules of the ISS.

Catastrophic collisions concern objects that are larger than 10 cm. In this case, the collision most likely results in the destruction of the spacecraft and the creation of additional debris. Catastrophic collisions are avoided by maneuvering the operating space object.

1.2.4 Consequences on Earth

The vast majority of human-made space objects are doomed to reenter on Earth. A reentry event is the return of an object from outer space where the atmospheric drag is negligible, to denser layers of the atmosphere until ground impact or complete ablation. This type of reentries is said to be destructive as the object is expected to be fragmented and burnt when reaching the ground by opposition to inhabited reentries from Soyouz capsules for instance. Two types of destructive reentries are possible: uncontrolled and controlled reentry. The uncontrolled (or natural) reentry happens when a non-operating object has been decaying from its original orbit due to residual atmospheric drag and without any human action. A controlled re-entry is triggered by an ultimate maneuver to deorbit the object toward a specific region on Earth. Note that the object is only controlled during the initial deorbiting maneuver, the rest of the reentry is completely uncontrolled.

On average one or two objects reenter on Earth everyday [6]. Occasionally, large objects such as rocket upper stages, large satellites, space stations reenter with masses exceeding several tons and cross-sections above $100m^2$ [6]. In this case, they can damage human assets, injure or kill someone, primarily if the reentry is not controlled. During a reentry, most of the object kinetic energy is converted into internal energy leading to fragmentation and ablation of the object such that small object may disintegrate completely before reaching the ground. For large objects, however, numerical studies and observations have shown that a significant part may reach the ground and present a hazard for human populations [34, 35, 36, 37, 38, 39]. The survivability of a fragment is its ability to survive to reentry (*i.e.*, to reach to ground). On the ground, the impact velocity can reach several dozens of meters per seconds of objects weighing up to 100 kg. For the Delta II, second stage reentry in 1997 over Texas, a 250 kg stainless steel tank was recovered near a farm [40]. In 2000, the same object reentered over Cape Town South Africa. For both reentries, the same kind of debris (pressure spheres and tanks) were recovered, leading to think that for each of 280 Delta II second stage launched until 2002, the same kind of debris impacted the ground [6]. In general, most of the surviving debris are not recovered as they land in an empty area or water. Even if the object falls over water and causes no damage upon impact, floating debris (propulsion tanks, pressure vessels) may be a hazard for ships or assets [41].

The casualty probability of heavily depends on the type of reentry. For instance, it is estimated that the controlled reentry of the MIR station (23 March 2001) had an almost 0 casualty probability because the reentry occurred over the Pacific ocean [6] whereas the Columbia shuttle reentry had an estimated casualty probability around 20 % [6]. The Columbia space shuttle reentered on February 1^{st} 2003 after a thermal system failure. The disintegration of the shuttle has been observed, and around 84 000 fragments were recovered (corresponding to 39 % of the initial shuttle mass) [42].

Except for catastrophic event such as the Columbia space shuttle, the personal probability of being hit is extremely low compared to other sources of risk. In [19], the risk is estimated at 10^{-12} per year to compare to the probability of being struck by lightning (10^{-7}) or having a home accident (10^{-5}) [1].

The second source of damage due to space object reentry is environmental. It concerns military satellites containing radioactive material (Radioisotope Thermo-electric Generator RTGs) used in LEO and GEO until 1976. Most of the RTGs are currently in LEO orbit where their life expectancy is about several hundreds of years. The RTG is made of radioactive plutonium ^{238}Pu which half-life time is around 24 000 years. Some RTG have already accidentally reentered for instance in 1964 after a launcher failure [1]. A military satellite RORSAT containing 30 kg

of radioactive uranium also accidentally landed on Canadian ground in 1977, leading to the first activation of the "UN liability convention" [6]. The reentry of space object raises other ecological concerns as residual propellant tanks and composite pressure spheres may damage natural habitat. Alloys such as Beryllium are considered as hazardous for local ecosystems [41].

1.3 Mitigation strategies and regulations

The proliferation of space debris calls for international regulations to protect future space activity. As of today, no international treaty aiming at reducing the number of space debris exists. On a state level, France has for example legislated for the regulations of spatial operations (Law n^o 2008-518 of the third of June 2008), and guidelines (soft laws) are suggested by several international agencies (NASA, ESA). Current international treaties, however, make states and International Governmental Organisations IGOs liable for their space activities in terms of human risk. They are free to carry space activities, but they are responsible for damages caused by the objects they launch as long as they exist and regardless of the ownership. Their liability is absolute and automatic without financial limit for injuries caused on the ground or to in-flight aircrafts [43].

While international laws are still pending to mitigate space debris proliferation, international guidelines have been under discussion since the '80s. In early 2000 the Inter-Agency Space Debris Coordination Committee (IADC) published the IADC Space Debris Mitigation Guidelines that served as the baseline for the UN Space Debris Mitigation Guidelines. In parallel, space agencies have been working toward technical requirements to meet those guidelines. Standards are defined in the ESA ADMIN/IPOL [30] to reduce the multiplication of space debris and ensure safer space activity for human populations. The objectives are to remove end-of-life satellites, bring down the in-orbit breakup risk, define protected regions and quantify the risk upon re-entry. The guidelines define protected areas such as the GEO and LEO or the Lagrange points that have particular economic and scientific value. Objects orbiting near the GEO ring or that may drift to GEO have to be re-orbited to a graveyard orbit 300 km above the GEO ring or 550 km below, or to heliocentric orbits [30]. End-of-life space objects should be passivated to avoid in-orbit breakups. Passivation includes emptying propellant tanks and discharging batteries [30].

Any object in LEO should reenter the Earth within 25 years [30] with limited on-ground risk. Different risk metrics characterize the on-ground risk. The casualty (or fatality) risk is the expected number of victim associated with the reentry of a space object. The casualty probability is the probability that the reentry kills at least one person. The casualty risk should be below 10^{-4} for a natural reentry according to the ESA guideline [30]. The second set of risk measures are the Declared Re-entry Area (DRA) and the Safety Re-entry Area (SRA). The DRA "should delimit the area where the debris should be enclosed with a probability of 99 %" while the SRA "should delimit the area where the debris should be enclosed with a probability of 99.999 %" [30].

The future launcher Ariane 6 will be compliant with the French Space Operation Act (FSOA) and ESA guidelines [20, 44]. This procedure includes the rocket stage passivation, its disposal mission, and re-entry. Furthermore, the on-ground risk associated with the reentering object will be quantified. In the case of a controlled re-entry, the SRA should not overlap with any territorial waters, of any states [20] and the maximum casualty probability should be kept below 2×10^{-5} [21]. For uncontrolled re-entries, the casualty probability should be below 10^{-4} [21]. A technical note demonstrating the compliance of the space mission to the French regulation have to be presented before any launch. In particular, it should include a robust estimation of the on-ground risk upon reentry.

1.4 Computer simulation for the on-ground risk estimation

The calculation of the on-ground risk as defined in [30, 21, 20] have to be performed a priori and before the object launch. According to [20, 44], it includes the estimation of the casualty risk, the casualty probability, the SRA and the DRA. Observation campaigns are too rare and expensive to provide a robust approach for predicting the risk. Hence the calculation of the risk has to rely on computer simulations. Computer simulations reproduce the reentry of a given space object using mathematical models that are solved numerically. The casualty risk, for instance, depends on the population distribution and the object impact location. The impact location is computed using numerical models that simulate the object reentry from its initial orbit to the ground impact.

Simulating the reentry of a space object is a complex task because it is a multi-physics problem. During a typical reentry, the space object leaves its initial orbit and decays to fall back on Earth. As its altitude decreases, the atmosphere slows down the object and heats its surface. The object usually breaks up into fragments that disintegrate in the atmosphere or reach the ground. Simulating this sequence of events is challenging because the phenomena involved are various. The simulation of the object orbital trajectory is utterly different from the simulation of the object trajectory in the denser regions of the atmosphere. Several mathematical models ranging from orbital mechanics, fluid mechanics, fluid-structure interaction or gas-surface interaction have to be coupled to predict the complete object reentry. The resolution of those models can be complicated to implement and computationally expensive to solve.

A practical solution to this complexity is to build a system of solvers. A system of solvers is a set of interdependent solvers, connected by coupling variables that are at the same time inputs of one solver and output of another one. In a system of solvers, each solver solves a specific aspect of the physics. One solver computes the orbital trajectory while another calculates the aerodynamic efforts. This approach permits to divide a complex problem into subsystems that can be managed by dedicated teams of experts in each domain. This approach still comes with a significant software development cost to interface the solvers and build a coherent, complete reentry simulator. A large panel of numerical models has been developed by space agencies (CNES, ESA, NASA) to predict a given reentry scenario and assess the risk associated. The models will be reviewed in details in Chapter 2. Most of the space object reentry simulators do not simulate the entire reentry event from the initial object orbit to the ground impact. Instead, the engineers have to launch a set of solvers manually and manage the interface between the solvers.

Computer simulations depend on mathematical models that are approximations of the phenomena we wish to reproduce. For instance, some phenomena such as the object breakup are still partially understood and therefore poorly modeled. Those approximations are necessary to maintain the computational cost and model complexity to a reasonable level, but they also introduce errors in the risk calculation. The errors associated with the computer simulations are discussed in details in Chapters 2 and 4. They can either come from the mathematical model that does not account for all the aspects of the physical problem, or from an inaccurate resolution of the mathematical model.

Additionally, the simulations are affected by other sources of uncertainties due to a lack of knowledge of the scenario. The initial conditions of the object reentry, for instance, may be partially known. These uncertainties and model errors need to be included in the simulation process to make robust predictions, especially for critical problems such as computing the on-ground risk. For this reason, the space agency guidelines suggest using statistical measures of the risk such as the DRA and SRA to account for the model errors and our lack of knowledge.

1.5 Uncertainty quantification and on-ground risk assessment

We refer here to Uncertainty Quantification UQ as the ensemble of methods permitting to estimate the influence of uncertainties in computer simulations and to quantify the variability of the numerical predictions due to the system uncertainty. UQ methods usually require numerous calls of the solvers under investigation and therefore are computationally demanding.

Some previous studies have investigated the influence of uncertainties in the on-ground risk [45, 46, 47, 48, 38, 49]. In [48], a simple parametric study has been performed to assess the importance of uncertainties on the on-ground risk. Each uncertainty is perturbed by a given offset from its nominal value, and the change in the estimated on-ground risk is measured. Three cases are considered: a satellite uncontrolled reentry, a Delta II type reentry and a satellite controlled reentry. In [50, 51, 52] efficient statistical methods are proposed to include uncertainties in the estimation of fallout regions. In [49], a Polynomial Chaos approach is used to propagate model parameters uncertainties through the solver. The use of a polynomial surrogate model allows for cutting down the computational cost of propagating uncertainties, compared to Monte Carlo approaches. However, they considered only a limited number of uncertainties, since getting a good polynomial approximation often requires a large number of runs to get convergence especially in high dimensions. In [38], a Monte Carlo (MC) and a Polynomial Chaos Expansion (PCE) approaches are used to propagate the uncertainties in the initial orbit characteristics (semi-major axis, right ascension of ascending node and the argument perigee) of a CubeSat and a standard satellite. The uncertainties are modeled as random variables allowing to compute the Sobol sensitivity indices and moments distributions. Unfortunately, the results are not entirely converged with the Monte Carlo approach (500 samples used) and for the PCE method [38] provides no error estimates associated with the surrogate model. Moreover, only a limited number of uncertainties could be considered due to the computational cost of one SCARAB evaluation.

While the analysis performed in [48, 38, 49] illustrate the significant influence of uncertainties in the risk predictions, they also lack efficient uncertainty propagation tools to accurately quantify the uncertainties in the on-ground risk and perform global sensitivity analysis. A primary objective of the thesis is to develop efficient UQ methods, specially adapted to the propagation of uncertainty in systems of solvers, and the calculation of the risk of human losses.

1.6 Challenges and objectives of this work

As mentioned before, laws and guidelines impose several technical constraints to space operators for controlling the reentry of space objects and providing a precise estimation of the on-ground risk associated with each reentry event.

The estimation of the risk necessarily relies on computer simulations of the complex phenomena associated with the reentry and more specifically, on multi-physics numerical models (structural, aerodynamics, trajectory, chemistry). The physics involved in this problem are various, complex, and even sometimes only partially understood. Uncertainties are numerous. And yet, the estimated on-ground risk must be fast to compute, robust and reliable to ensure safe and sustainable space operations. Some of the main challenges to tackle are the following:

1. **Assess the reliability of the numerical models**: The numerical simulation of the fluid-structure and gas-surface interaction phenomena in combination with the trajectory evolution is essential to predict the forces acting on the object, the energy transfer, and the potential fragmentation of the object. From a modeling point of view, this phenomenon is extremely rich, and only little data exist for properly calibrating the models. Additionally, a

high-fidelity simulation during the whole trajectory is extremely computationally demanding. A classical approach is to use simplified models, which induce additional sources of errors. Considering the numerous sources of uncertainties coming from the model approximations, a reliability assessment of the numerical prediction is primary to trust the estimation of the risk.

2. **Enhance robustness of the breakup models**: The fragmentation process has critical consequences on the object aerodynamics and the rest of the trajectory but is also the least understood phenomenon and the most complex to predict. The breakup is defined as the fragmentation of the object into two or more fragments and is the combined results of intense heat flux, aerodynamic efforts, and tank or battery explosion. Accurately modeling those phenomena and their interactions is exceptionally challenging. Even the most advanced breakup models solely model the thermal aspects and sometimes employ basic criteria to model the aerodynamic load effects. The breakup model defines the number of fragments and their characteristics. It has a foremost influence on the predicted on-ground risk.

3. **Model and propagate uncertainties in reentry predictions at a reasonable computational cost**: Space object reentry simulators are affected by uncertainties that have to be identified, modeled and propagated through the simulator to obtain a robust estimation of the risk. This estimation should be performed at limited computational cost, thus avoiding classical sampling methods. Currently, uncertainties are not considered systematically when predicting space object reentry due to the prohibitive cost of standard uncertainty quantification methods used with multi-physics problems like this one.

4. **Identify the uncertainties that affect the most the estimated on-ground risk**: Identifying the parameters or model errors that induce the most uncertainties in the model is crucial to work on reducing the uncertainties in the prediction and improve the model reliability.

In this thesis, we propose several contributions that provide partial answers to these challenges:

- **Construct a numerical model for reentry prediction**: In this work, we propose two reentry simulators. The first one is presented in Chapter 3 while the second one is presented in the Appendix. The simulators can simulate the entire object reentry event by coupling several solvers together into systems of solvers. Moreover, the solvers are interfaced and coupled in a unique framework to allow automatic, parallel launches from a single user.

- **Proposal of breakup models under uncertainty**: We propose a novel approach to model breakup. We model it as a random event to account for our inability to accurately simulate the complex sequence of events that lead to breakup and fragment release. Two breakup models are developed, one for each reentry simulator.

- **Develop efficient uncertainty propagation tools for Systems of Solvers** : We present two original solutions to propagate the uncertainties in the system of solvers and reentry simulators: i) An original framework for efficiently constructing surrogate models of systems of solvers (see Chapter 4). The framework also includes efficient active learning strategies to improve the performance of the surrogate model adaptively. The framework developed is general and can be applied to a large set of industrial problems. ii) A novel formulation of the fragment survivability that permits to construct surrogate models. We view the survivability assessment as the joint prediction of a classifier and a continuous function. We derive an original surrogate model strategy assisted with an active learning strategy to derive the on-ground risk distribution efficiently.

- **Characterization of the sources of uncertainties in the SoS**: In chapter 2, we investigate the models available to predict the reentry of a space object, and we identify the sources of uncertainties and the errors in the models. We then propose to model them as random variables and to study their influence on a complete upper stage reentry problem. We carry an analysis of the model output sensitivity to the uncertain inputs. We are able to rank dozens of uncertainties according to their influence on the output uncertainties and to formulate recommendations for reducing the output uncertainties.

- **Proof-of-concept of the developed UQ tools with application to a reentry simulator for predicting the reentry of an upper stage under uncertainty**: We develop a software tool for reentry prediction under uncertainty, combining the different contributions of this work: the reentry simulator proposed in Chapter 3 with the probabilistic breakup model, and the surrogate modeling strategies. The software tool is applied to perform a physical analysis of the uncertainties based on rigorous sensitivity analysis and the robust estimation of the quantity of interest (breakup flight conditions, on-ground risk). This analysis is performed at a very reasonable computational cost.

1.7 Outline

Chapter 2 In this chapter, we present the physical context of this work. We review in details the main flow properties (hypersonic, high enthalpy), the gas surface interaction phenomena, the heat flux, the ablation phenomena, and atmosphere characteristics.

Chapter 3 In this chapter, we build a reentry simulator using models developed by ArianeGroup (AG). It is composed of several heterogeneous solvers that had to be included in a common framework. We provide some details about the integration of several solvers into a general python framework that minimizes the user interventions. This interface permits parallel launches in view of performing uncertainty quantification. We then critically investigate the main assumptions used in object reentry models and explain how they can induce prediction errors. We use this discussion to expose an overview of the uncertainties involved in the prediction of a space object reentry. We discriminate two classes of uncertainties: the model errors and the scenario uncertainties.

Chapter 4 This chapter introduces the uncertainty quantification methodology. First, we review the existing uncertainty quantification methods and in particular surrogate modeling approaches and design of experiment strategies. Second, we present a new approach for constructing a surrogate model of a directed system of solver where each solver is approximated with a Gaussian Process (GP). The surrogate model is a system of Gaussian Processes (SoGP), and like a Gaussian Process, it spans a predictive distribution of the output. We show how the predictive variance of the SoGP can be decomposed into contributions from each of the composing GPs. This decomposition is then used to propose three different active learning strategies adapted for SoGP. One of the active learning strategies can identify the least reliable GP of the SoGP and to suggest points for improving the quality of the prediction. The performance of the SoGP framework is compared to a single GP on several analytical test functions and a simplified reentry problem.

Chapter 5 In this chapter, we apply the SoGP framework to construct a surrogate model of the simulator developed in Chapter 3. The SoGP is used to predict the breakup flight conditions

(position and velocity of the object). We show the advantage of using the SoGP framework from Chapter 4 on a full-scale industrial problem as it outperforms the single GP by one order of magnitude. Based on the constructed surrogate model, we demonstrate the advantage of using a probabilistic breakup model for performing robust reentry predictions. We can include a large number of uncertainties (28) and to perform a sensitivity analysis at minimal computational cost using the SoGP framework.

Chapter 6 This chapter proposes a new surrogate model construction strategy adapted to the survivability prediction of the fragment. The output of the survivability solver is composite and cannot be approximated well by a GP and hence by the SoGP framework. In this chapter, we propose an original formulation of the survivability problem and build a surrogate model composed of a classifier and a GP. Furthermore, we suggest an active learning strategy based on the predictive variance of the constructed surrogate model. The performance of the surrogate model and the active learning strategy are tested on several analytical cases and a preliminary reentry problem with a single fragment.

Chapter 7 Based on Chapters 4 and 6, we can construct a surrogate model for the entire reentry simulator. We combine the SoGP framework from Chapter 4 with the survivability surrogate model from Chapter 6 to build a surrogate model of the reentry simulator to estimate the on-ground risk under uncertainty. The excellent performance of the surrogate model strategies permits to include several dozens of uncertainties in the analysis and to perform a thorough sensitivity analysis of the entire system.

Chapter 8 In this final chapter, we summarize the findings of this work and formulate possible future developments based on the results of this work.

Appendices In Appendix A, we present in details the ArianeGroup solvers used in the reentry simulator presented in Chapter 3. In Appendix B, we propose a higher fidelity simulator compared to the simulator constructed in this study. It features better modeling of the object thermal response to the external flow and a more advanced probabilistic breakup model. A synthetic comparison between the two models in a deterministic setting is also proposed.

Chapter 2

Space Object Reentry

In this chapter, we illustrate the main physical phenomena that occur during the reentry of a space object. Furthermore, we provide different definitions of risk according to the main guidelines.

2.1 Some definitions

A reentry event is the return of an object from outer space where the atmospheric drag is negligible to denser layers of the atmosphere until landing or complete ablation. By convention, the reentry usually begins at an arbitrary altitude 120 km where the atmospheric density is around 10 million times smaller than the sea level atmosphere density. The objects considered in this work are non operating, human-made space objects such as rocket bodies or satellites but do not concern inhabited flights. This type of reentry is denoted as destructive as the object is expected to be fragmented and burnt. Two types of destructive reentries are possible: uncontrolled and controlled reentry. The uncontrolled reentry happens when a non-operating object has been decaying from its original orbit due to residual atmospheric drag. A controlled re-entry is triggered by an ultimate maneuver to deorbit the object toward a specific region on Earth. In the case of a controlled reentry, the French Space Operation Act requires that the SRA do not impinge upon any territorial water or lands and that the casualty probability and the expected number of victims be lower than 2×10^{-5}. In the case of an uncontrolled reentry, it must stay below 10^{-4}. [44, 21]

Whether the reentry is controlled or not, a reentry event can be decomposed into the following sequence of events (see Fig. 2.1).

- Upper atmosphere trajectory (120 km -90 km altitude): During this part of the reentry, the atmosphere is around a million times less dense than at sea level. The aerodynamic forces are limited, and the object orientation remains stable. The flight regime is hypersonic in a rarefied flow.
- Pre-breakup trajectory (90 km altitude until breakup): During this phase, the atmosphere effects become more and more important such as the object starts tumbling and is heated by convective, reactive and radiative heat flux.
- Breakup or fragmentation: At some point in the trajectory the object will certainly break up. Optical observations of the Columbia space shuttle, ATV reentries or the MIR space station reentry suggest that the object fragmentation is not a single catastrophic event but a sequential breakup cascade. After the sequence of breakups, the fragments originating from the initial object keep falling back to Earth.

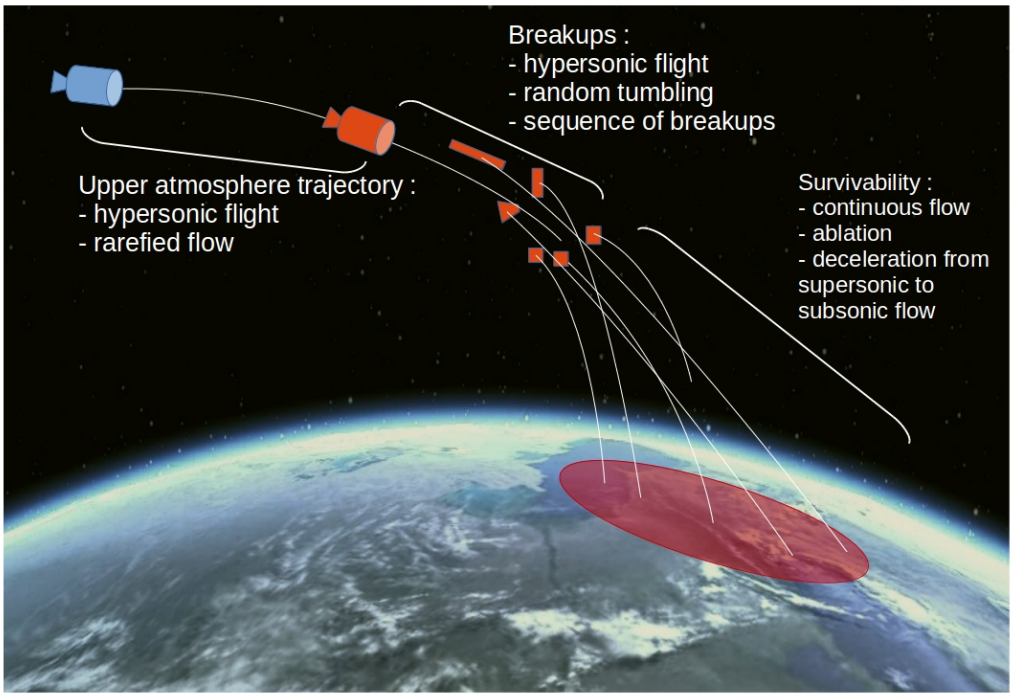


Figure 2.1: Simplified representation of an upper stage reentry.

- Survivability: At this point, no further breakup is expected, but each fragment is slowed down and ablated depending on its shape and material composition. Some of the fragments completely disintegrate while some of them survive and reach the ground. Around 20 km, the surviving fragments experience a free fall at subsonic velocity depending on their aerodynamic properties. Depending on the impact locations of the fragments, the expected number of victims and the probability of having a casualty can be computed.

This chapter describes the main physical phenomena occurring during a reentry (Section 2.2). Then, definitions of risk are illustrated in Section 2.3.

2.2 A multi-physics problem

We briefly illustrate here why the reentry of a space object is a multiphysics phenomenon. We introduce the atmospheric model, and then we describe several aspects of the reentry of a space object in terms of aerodynamics, gas-surface interactions, structural thermomechanical response, and object breakup.

2.2.1 The role of the Atmosphere

The atmosphere composition plays an essential role in the estimation of the aerodynamic forces. Its characteristics vary drastically depending on the altitude. The mean pressure and density decrease with altitude while the temperature follows more complex variation due to sunlight absorption in the stratosphere (15 to 50 km) and the thermosphere (above 80 km). In Fig. 2.2, the variations of temperature, pressure, and density are represented. The Earth atmosphere is decomposed into four layers. The troposphere (0 to 15km), the stratosphere (15 to 50 km), the

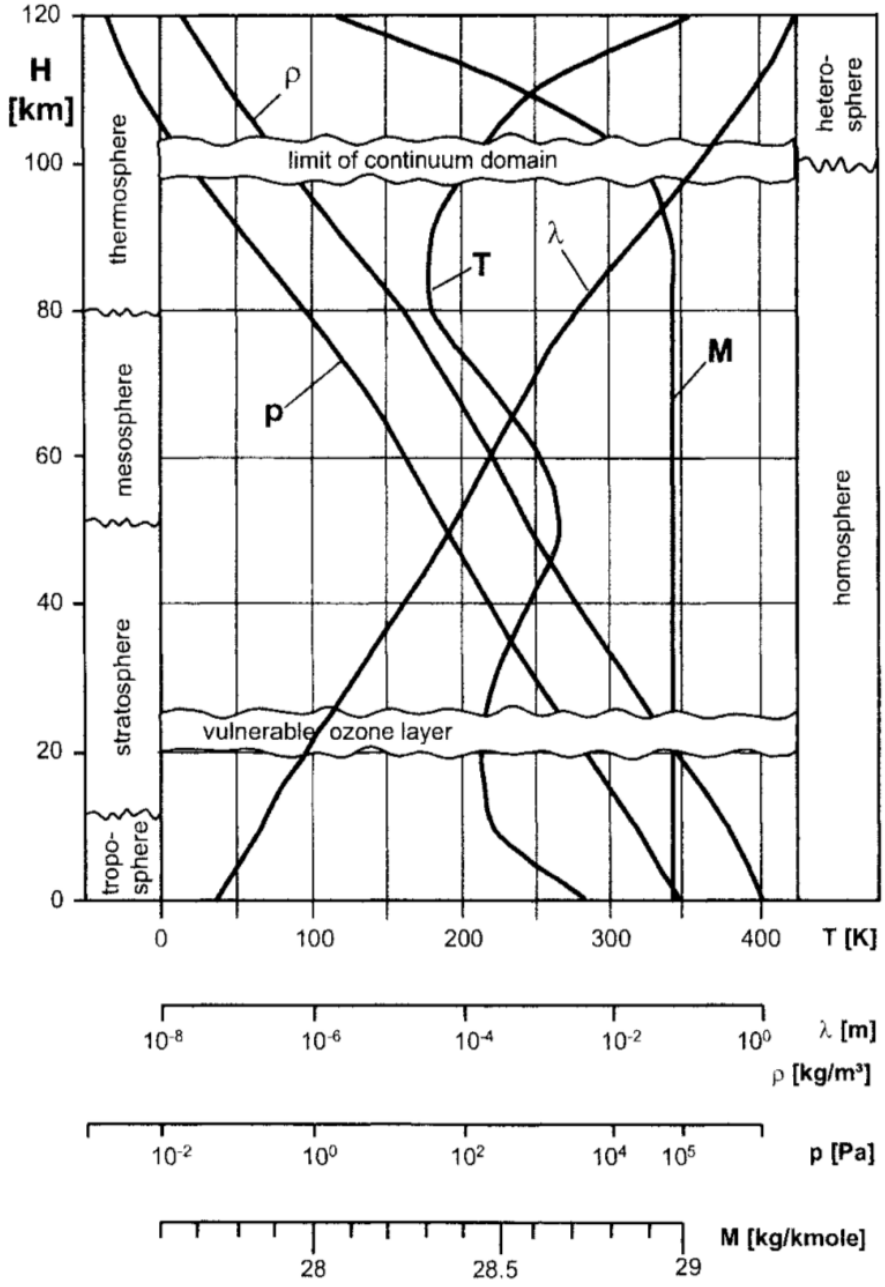


Figure 2.2: Atmosphere characteristics (from [3]

mesosphere (50 to 80km), the thermosphere (80 km to 1000 km) and the exosphere (1000 to 10 000 km).

In the exosphere, the molecules follow ballistic trajectories, and in the thermosphere, the free molecular flow ranges from several meters to kilometers. The atmosphere does not only depend on altitude but also on solar activity, season and time of the day. In the troposphere, the air is turbulent and strong wind effects can significantly affect the trajectory of fragments that, at this point of their reentry, have subsonic velocities.

The thermosphere and exosphere are highly turbulent and complicated to model because of their sensitivity to solar activity and Earth geomagnetic activity. In the case of uncontrolled reentry, the space object may remain several hours or days in the thermosphere before re-entering

on Earth. Variations in the density profile influence the drag force and in turn the trajectory. Hence the variability of the atmosphere can have a significant influence on the re-entry window. At lower altitudes, the density profile also affects the convective heat flux and in turn the object survivability.

Several atmosphere models have been developed to model the atmosphere. The first one, US62, uses an exponential pressure variation model and models air like a perfect gas [53]. The model has been improved in US76 and MSIS-2000 [54]. In its latest version, MSIS-2000 includes seasonal variation, hour variation, solar activity, geomagnetic activity, and location. Compared to earlier versions, MSIS-2000 present significant differences for altitude above 120 km. Nevertheless, it does not include any wind modeling that can significantly affect the trajectory below 20 km altitude.

2.2.2 Aerodynamics

From initial reentry conditions to ground impact, a space object goes through a large set of different flow regimes that affect its aerodynamics and heat transfer with the atmosphere. In this section, we discuss the different flow regimes. We investigate in particular three flow characteristics: the flow regime characterized by the Knudsen number, the speed regime characterized by the Mach number, and the dynamic behavior described by the Reynolds number.

Density flow regime

At 120 km altitude where the space object reentry begins, the atmospheric density is a million time lower than at sea level where reentry ends. When the density is extremely low, the air cannot be seen as a continuous medium since the distance between molecules becomes comparable to the space object size. The Knudsen number is defined as:

$$K_n = \frac{\lambda}{L} \tag{2.1}$$

where λ is the molecular mean free path which is the average distance a molecule travels between two collisions, and L is the object characteristic length. The Knudsen number allows to define three regimes:

- The free molecular flow is defined for $Kn > 1$. In this case, the gas density is so low that only a limited number of molecules impinge upon the surface of the space debris. Classical fluid mechanics models cannot be directly applied. On the other hand, results from gas kinetic theory can be used to compute the aerodynamic force and heat transfer [55].

- The rarefied flow or transition regime is commonly defined between $Kn < 1$ and $Kn > 0.03$ [4]. As for the free molecular flow, the continuity assumptions are not valid. The rarefied flow is characterized by slip conditions the surface of the space object. The flow velocity at the vehicle surface, for instance, is not 0. Similarly, the flow temperature at the object surface is not equal to the surface object temperature. In general, the Navier-Stokes equations from the continuum regime are not valid. Instead, the Boltzmann equation has to be solved. In the special case where $Kn < 0.2$, the Navier-Stokes equations can be applied with specific slip conditions at the object surface. [4]

- The continuum flow is defined for $Kn < 0.03$. In this regime, the continuum assumptions are valid, and the Navier-Stokes equations can be used to characterize the flow.

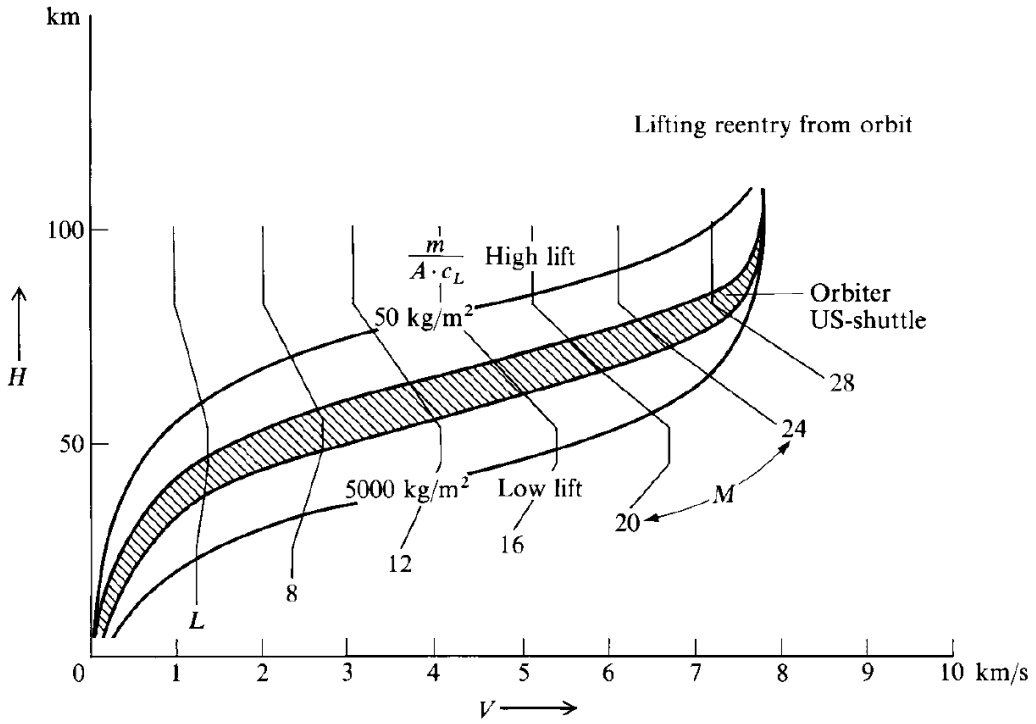


Figure 2.3: Velocity-altitude map with superimposed lines of constant Mach number [4]

The speed regime

The different flow regimes are characterized by the Mach number M :

$$M = \frac{v}{c}, \tag{2.2}$$

where v is the free-stream velocity and c is the speed of sound. The Mach is the square root of the ratio between the flow kinetic energy and the flow internal energy. Depending on the Mach number the following regimes can be identified :

- Hypersonic flow: conventionally defined for Mach numbers above five although some hypersonic flow characteristics can be observed at lower Mach numbers. The hypersonic flow is encountered at the early stages of the re-entry.
- Supersonic flow: the supersonic flow is faster than the speed of sound and characterized by the creation of a shock in front of the object (Mach numbers between 1 and 5).
- Subsonic flow: for Mach number below 1. During reentry, the subsonic flow is observed toward the end at low altitudes and after the main object fragmentation.

The evolution of the Mach number and change in the flow regime along a re-entry trajectory are presented in Fig. 2.3. During reentry, the hypersonic regime is dominating from 120 km to 50 km and often after breakups usually occur during the hypersonic flow phase. In the rest of this paragraph, we give an overview of the main characteristics of hypersonic flows.

Hypersonic flows [4] feature five significant characteristics, which are described in the following:

1. **Thin shock layer** For hypersonic flights, the shock layer between the object and the shock reduces in thickness. For oblique shocks, the flow density increases with the Mach

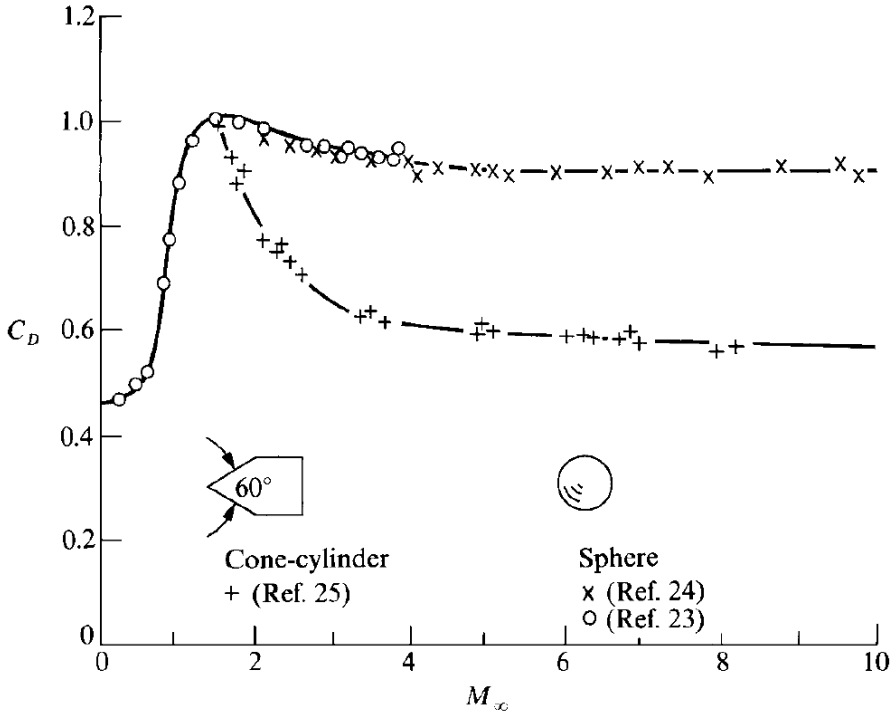


Figure 2.4: Measured drag coefficient for a sphere and a cone-cylinder [5]

number such that the mass flow is concentrated over a smaller volume when flow becomes hypersonic. [4]

2. **Mach number independence** Hypersonic flow characteristics are independent of the Mach number, as shown in [4] at the limit $M \longrightarrow \infty$ the flow governing equations become independent of the Mach number. In the case of a space object reentry, for hypersonic flight, the aerodynamic forces are independent of M. This trend is observed experimentally in Fig 2.4. Note that the Mach independence regime is reached at different rates depending on the object shape. For blunt bodies, the Mach number independence regime is reached for relatively lower Mach numbers than for sharp bodies. See for instance the difference between the sphere and the cone-cylinder on Fig. 2.4.

3. **Viscous boundary layer** As discussed earlier, hypersonic flows have high kinetic energy that is converted into gas internal energy in the boundary layer. This conversion is called viscous dissipation. This effect leads to an increase in the boundary layer temperature and the gas viscosity increases and in turn it thickens the boundary layer. In particular, we have that the boundary layer thickness scales as [4] :

$$\delta \propto \frac{M_\infty^2}{\sqrt{Re_x}} \tag{2.3}$$

where M_∞ is the freestream Mach number and Re_x is the local Reynolds. In hypersonic flow M_∞ becomes large and so does δ. The increased boundary layer changes the pressure distribution and the aerodynamic forces that apply to the space object. In some cases, the boundary layer and the shock layer may have a similar size.

4. **Reacting boundary layer** As discussed in the previous paragraph, at the shock, the kinetic flow energy is converted into gas internal energy, namely vibrational and rotational

energy. The flow temperature increases and effects specific to high energy gases can be observed. The specific heats become functions of temperature and the specific heat ratio γ is not constant. The increase in vibrational energy leads to molecular dissociations into atoms, ions, and electrons. At 800 K molecular vibrational modes start being excited. At sea level, Dioxygen dissociates as soon as 2000 K and is completely dissociated at 4000K. For Nitrogen, it is entirely dissociated at 9000K. At 9000K, molecular ionization can occur. Figure 2.5 summarizes the velocity and altitudes for which air molecules will be decomposed.

The flow right after the shock is not at equilibrium. While rotational and translational equilibrium is usually reached within a few collisions, vibrational equilibrium is reached after the order of 100 000 collisions [6]. The vibrational relaxation time depends on the number of collisions a molecule experiences in the shock layer. It depends on the flow field velocity and density.

We define the vibrational relaxation time τ_{vib} as the time to reach equilibrium after crossing the shock and τ_{res} the time a molecule resides in the shock layer, three regimes can be defined. If $\tau_{vib} \gg \tau_{res}$, the flow is vibrationally frozen, that is the flow vibrational energy remains constant after the shock. This case can be encountered at the early stage of the reentry when the flow density is low. If $\tau_{vib} \sim \tau_{res}$, the flow is not at equilibrium, the vibrational energy varies in space and depends on the particle type [56]. If $\tau_{vib} \ll \tau_{res}$, the flow is at equilibrium. The vibrational temperature is the same as the translational temperature. Similar remarks can be formulated for the chemical reaction time τ_{ch}. After the shock, the molecules are decomposed into atoms and ions that recombine in the shock and boundary layer. If $\tau_{ch} \gg \tau_{res}$, then the flow is chemically frozen, the flow chemical composition remains constant in the shock layer. Almost no molecular recombination takes place in the shock layer except possibly at the surface of the body. If $\tau_{ch} \sim \tau_{res}$, flow is out of equilibrium and the molecular composition varies in space. If $\tau_{ch} \ll \tau_{res}$, the shock layer can be considered at local chemical equilibrium and the gas state equation gives the chemical composition.

In the boundary layer, the gas composition is also affected by the catalytic properties of the object surface that accelerates the molecular recombination rate at the object surface. Finally for very high temperature flows around 10 000 K (Apollo 11 reentry was about 11 000 K at Mach 36 [4]), the radiation emitted and absorbed by the gas become non-negligible and locally affects the flow energy [56].

Dynamic flow regime

The Reynolds number characterizes the dynamic regime of the flow:

$$Re = \frac{\rho v_\infty L_{ref}}{\mu} \tag{2.4}$$

where ρ is the fluid density, v_∞ is the free stream velocity, L_{ref} is the object reference length, and μ is the dynamic viscosity. The Reynolds number is the ratio between the inertia forces and the viscous forces. Depending on the Reynolds number, the flow regime can be laminar or turbulent. A turbulent flow features "random variations with time and space" (Hinze 1975) [57], it is inherently three dimensional and time-dependent. The space length scales of variation of turbulent flows can be extremely small. Fig. 2.6 shows the turbulent wake trail of a bullet at Mach 1.5. During a space object reentry, the object usually experiences substantial changes in the Reynolds number.

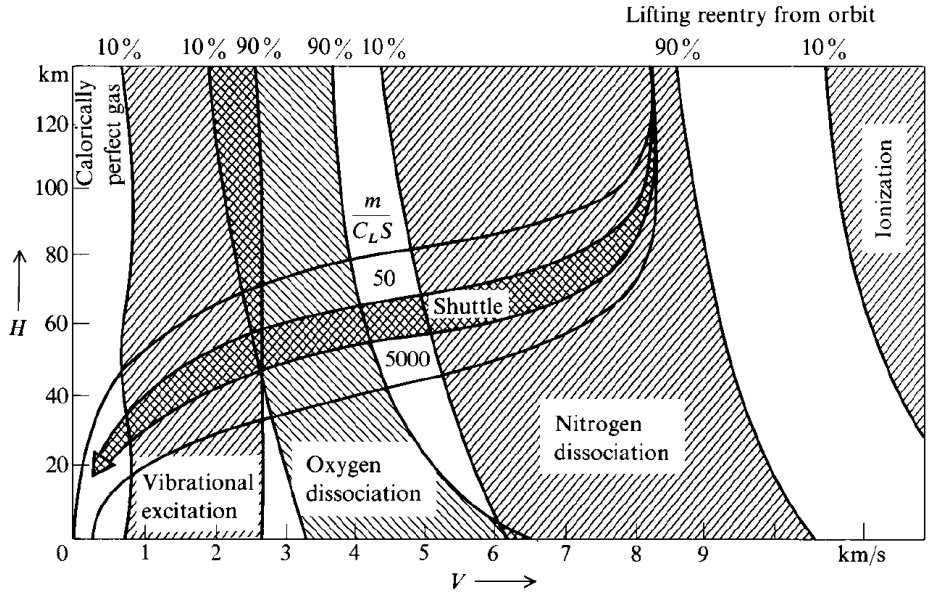


Figure 2.5: Molecular dissociation, vibrational excitation and ionization function of altitude and velocity from [4].

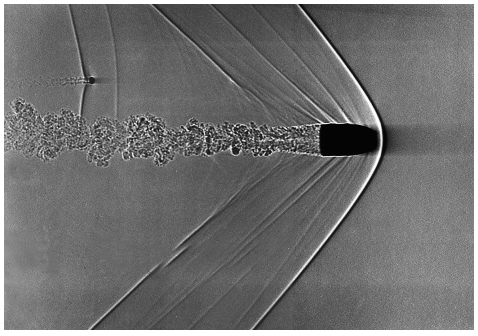


Figure 2.6: Photography of a bullet moving at Mach 1.5 (source NASA)

As shown in Fig. 2.7, the early stage of a reentry features high Mach numbers and low Reynolds number where a viscous boundary layer is likely to be observed. At lower altitudes, the flow transitions to a turbulent, flow. As an illustration, Fig. 2.7 shows the transition from laminar to turbulent for a 10-meter object at Reynolds equal to 10^6. The laminar flow at the leading edge of the object transitions into a turbulent further as it moves downstream. The transition to turbulence in hypersonic is still not perfectly understood. It depends on the transition Reynolds number that in turn depends on a large number of flow parameters [4]. Among them, the Mach number at the outer edge of the boundary layer (M_e). Studies found that the transition Reynolds number increases with the Mach number for $M_e > 4$, implying greater flow stability for hypersonic flows than for slower flow. Other key parameters such as the angle of attack, the environment, the wall temperature, and the object geometry also affect the turbulent transition [4].

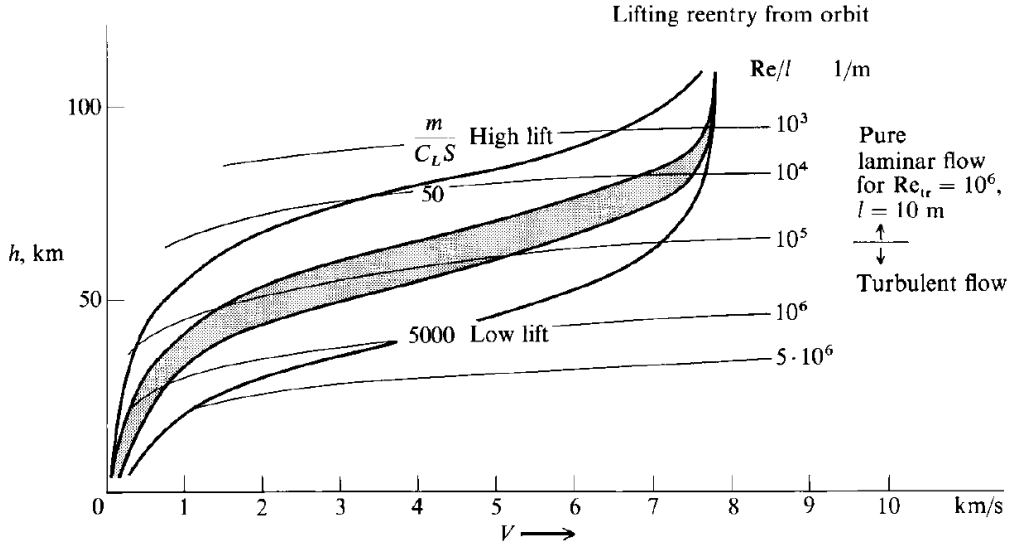


Figure 2.7: Reynolds number evolution with reentry trajectories. (from [4])

2.2.3 Gas-surface interactions

In this section, we look at the interaction between the gas and the object surface. As previously discussed, viscous dissipation in the boundary layer can lead to very high gas temperature, molecular dissociation, gas radiation, and wall radiation. As a result, the gas interacts with the object surface through molecular recombination, convective heat transfer, radiation, and ablation. All those interactions consist of exchanges of energy or mass.

- **Convective heat flux** The convective heat flux corresponds to the heat exchange between the hot gas and the cold surface, due to temperature differences. The convective heat flux depends primarily on the enthalpy difference, the flow density and dynamic behavior of the flow.

- **Wall radiative heat flux** The radiative heat flux comes from the object surface or the shock layer. A black body radiates energy following the Boltzmann law:

$$q_r = \epsilon \sigma T^4, \tag{2.5}$$

where ϵ is the body emissivity, σ the Stefan-Boltzmann constant and T the body temperature. This effect is called radiative cooling. It is all the more important that the object temperature is high. For high fusion temperature materials such as titanium, the radiative heat flux may become a major cooling effect, leading to the survivability of the object.
The gas in the shock layer also radiates energy. This effect depends on the gas composition and temperature. The radiative gas effect becomes non-negligible at around 10 000 K. In radiative flows, the flow exchanges energy with other flow regions or the body. This phenomenon creates complex coupling effects between fluid elements [4] depending on the gas properties. If the gas is transparent, it radiates energy but does not absorb it. On the contrary, self-absorbing gas emits and absorbs radiation. Gas radiative heat flux is significant for high-speed reentries above 10 km/s and large objects (around 5 m) [4]. For the Apollo reentry, estimations showed that the gas radiation heat flux accounted for 30 % of the total received heat flux.

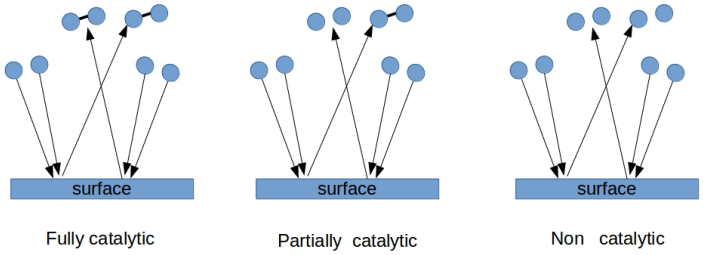


Figure 2.8: Schematic representation of the surface catalysis phenomenon for three types of materials : fully catalytic, partially catalytic and non catalytic

- **Molecular recombination**

  The sudden conversion of the flow kinetic energy into internal energy leads to the molecular decomposition of air molecules into nitrogen and oxygen atoms. The object surface can act as a catalytic medium facilitating molecular exothermic recombination. The energy released by the molecular recombination occurring at the surface of the space object contributes to the heat flux transferred to the object. See Fig. 2.8.

  The molecular recombination rate depends on the catalytic properties of the material. Materials such as quartz are assumed to be non-catalytic. The probability of observing molecular recombinations between two impinging atoms is extremely low [58].On the contrary, fully catalytic materials feature a recombination probability close to one. Silver is often considered as a fully catalytic material [58]. In between, the partially catalytic materials have a recombination probability significantly lower than one, which is the case for most materials used in space object reentry. Note that the material catalytic property changes during the reentry as the surface chemical composition varies over time due to ablation and oxidation. The molecular recombination heat flux also strongly depends on the chemical composition in the boundary layer, if the flow is frozen and most of the atoms do not recombine in the boundary layer, more recombination reactions take place at the surface. If the flow is at equilibrium, depending on the flow enthalpy, pressure and density, a significant number of atoms may have already recombined when reaching the object surface.

- **Surface oxydation**

  In addition to molecular recombination, the material itself may react with the oxygen atoms into exothermic oxidation reactions that release energy at the surface of the object. In Fig. 2.9 the surface of titanium oxide obtained in the PROMES furnace is represented. The oxidation reaction rate depends on the surface regularity and the flow conditions [6] and if an oxide layer is already formed.

2.2.4 Structural and material behavior upon reentry

Space objects are composed of various components with very different materials. The material composition of a space object depends on its primary purpose (observation, broadcast, scientific

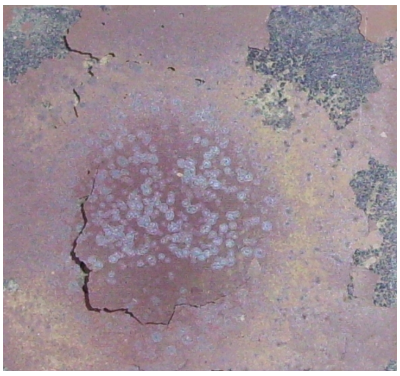


Figure 2.9: Oxidized surface titanium (from [6])

experiment, propulsion). The composition and the structure of a reentering object have a significant influence on the object thermal response and the breakups. In this section, we focus on two types of materials with very different characteristics and behaviors: the metal alloys and the composite materials that tend to be used more and more for their reduced weight and excellent mechanical properties.

Metals and alloys A solid-state metal or alloy has a crystalline structure. That is its atoms are arranged in a highly organized microscopic structure that repeats itself in all directions of space. The compact organization and well-organized structure of the metal ensure its structural stability while flaws in the crystal affect its thermal and mechanical properties [6].

Aluminum and Titanium are the most used materials in the aerospace industry. Aluminum alloys are used for large structures such as upper stage tanks and satellite buses. They offer a good compromise between weight and strength. They are also characterized by low fusion temperature and tend to disintegrate completely upon reentry. Titanium is used for joints between aluminum plates or parts exposed to intense heat flux and pressure such as pressure spheres or gas chambers and nozzles. Titanium alloys have a better mechanical and buckling resistance up to 750 K than aluminum but remain likely to fail at the welding between two parts [6]. More recent designs use new materials such as Beryllium that has excellent mechanical properties and low weight but it remains costly and toxic. [6]

Concerning the behaviour for high-temperature flow, the surface of an alloy can catalyze molecular recombinations of nitrogen atoms, oxygen atoms, and electrons. The heat released by the exothermic reactions directly contributes to the heat transfer from the gas to the material. The heat transfer due to molecular recombination depends on the chemical composition of the flow at the surface and microscopic structure at the surface of the object. In addition to molecular recombination, oxidation reactions can occur. The reaction involves an oxygen atom and metal atoms or oxide and yields an oxide. Not only this exothermic reaction significantly contributes to the heat transfer from the gas to the object, but it also changes the material properties at the surface such as emissivity, fusion temperature, and catalytic properties. The formation of alumina at the surface of an aluminum material changes the melting point from 800 K to 2345 K. In [59], the emissivity of the titanium is shown to increase from 0.2 to 0.8 when oxidized in air plasma. A oxidized titanium plate is represented in Fig. 2.9. The formation of oxide is usually solely located at the surface on a thin layer, but the increased fusion temperature and emissivity can affect the radiated heat flux.

Under a high enthalpy flow, the material is subject to ablation. For metals, it is generally assumed that when the surface of the object reaches fusion temperature, the melted matter is

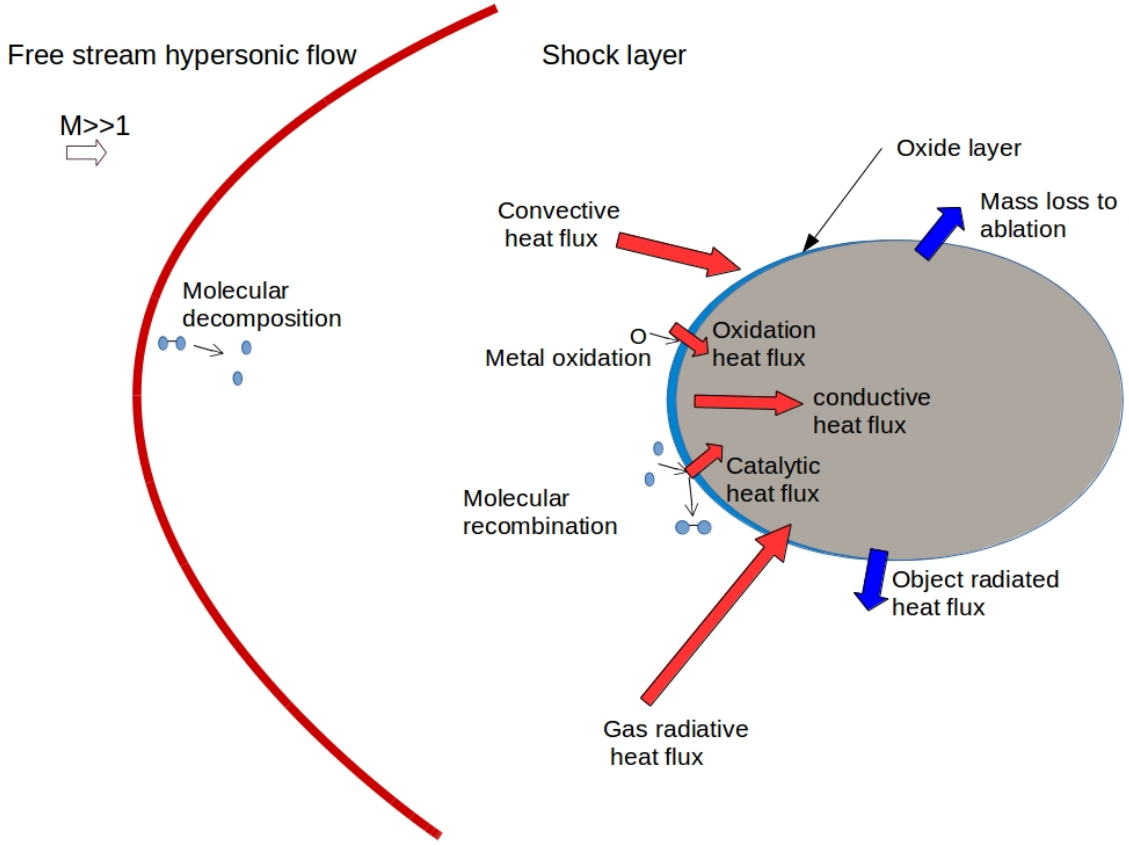


Figure 2.10: Schematic representation of interface phenomena between a high enthalpy flow and a metallic material

swept by the flow. Formation of the oxide layer complexifies the phenomena. The oxide fusion temperature is much higher than the metal fusion temperature, and in some case, the metal under the oxide layer may melt before the oxide layer reaches fusion temperature [8]. In [6], the titanium oxide layer observed in 2.9 is non-protective, it does not prevent further oxidation and it even facilitate ablation [6] while for aluminum it creates a protective skin of alumina with very different material characteristics (emissivity, fusion temperature, etc.) [8]. In this case, the ablation metal of the metal may be delayed depending on the aerodynamic forces.

Composite materials A composite material is composed of at least two non-miscible materials with different mechanical and physical properties. Contrary to alloys, composite materials remain heterogeneous at the microscopic scale. They are typically composed of a reinforcement that makes the frame and a matrix that surrounds the reinforcement material (see Fig. 2.11). The reinforcement material defines the mechanical properties of the material. The reinforcement is usually made of fibers (metal, glass or polymers) that can be positioned all in the same direction or randomly. Common types of fibers are carbon fibers. Their fibers have excellent resistance to traction but not to compression [6]. The matrix is a soft component made of silicon (for instance), enriched with additives (for instance rubber). Ceramic matrix composite CMC are primary choices for thermal protection material. CMC have a high resistance to thermal and mechanical loads. Carbon fiber reinforced polymer CFRP have been recently used in aerospace

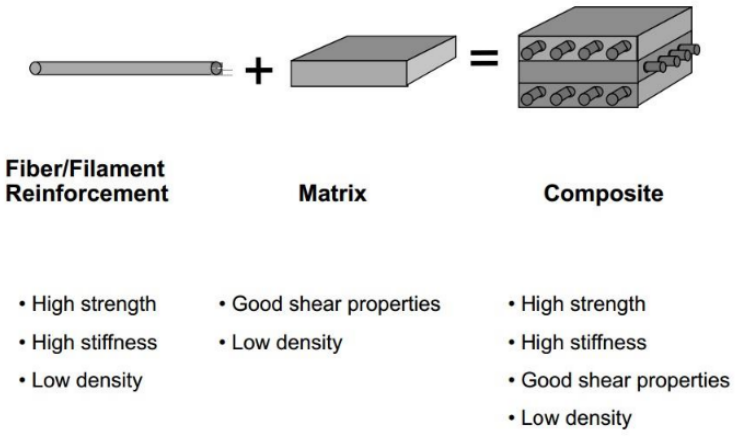


Figure 2.11: Composition of a composite material from [7]

applications. Its main advantages are a very low weight coupled with high mechanical resistance of the fiber. Composite materials are extensively used in tanks, pressure tanks in satellites and upper stages (see Fig. 2.13).

Concerning the behaviour for high-temperature flow, composite materials can act as a catalytic surface for molecular recombination that contributes to the received heat flux. Additionally, the response to ablation differs from the alloys. The composite material can be decomposed into three distinct zones. The virgin zone, the pyrolyze zone, and the char zone. Initially, only the virgin zone is present. The flow heat flux leads to an increase in temperature until 500K-600K. At this temperature, the pyrolyze reactions start. The pyrolyze corresponds to the decomposition of the resin filling into a gas. It yields the following effects [60]:

- This endothermic reaction acts as a heat sink that protects deeper layers of the material.
- The pyrolyze reaction also creates outgassing that blocks the incoming flow and reduces the convective heat flux .
- The pyrolyze zone becomes porous without the resin and broader as the heat diffuses in the material and the temperature increases.

In the char zone, when the temperature increases, the carbon residual can be oxidized by impinging oxygen atoms or molecules. As for the metal, this reaction generates additional heat flux. The carbon atoms may also react with the silicate fibers at a temperature around 1300 K. The changes in the chemical composition of the composite material significantly affect their thermal and mechanical properties. The ablation process for the composite materials such CFRP are still not well understood whether the material is sublimated or if the char zone becomes too fragile and it is swept by the flow. It is generally accepted that charred materials in CFRP start being sublimated around 3000 K, a temperature that is rarely reached during space debris reentries [6]. Hence, the ablation phenomena are most certainly due to the erosion of the char zone [60].

2.2.5 Breakup analysis

Breakups are almost sure events during a reentry. In [61], two types of breakups can be identified: the low energy breakup and the high energy breakup. Low energy breakups almost always take place, they are due to the structural failure of the object while the high energy breakups

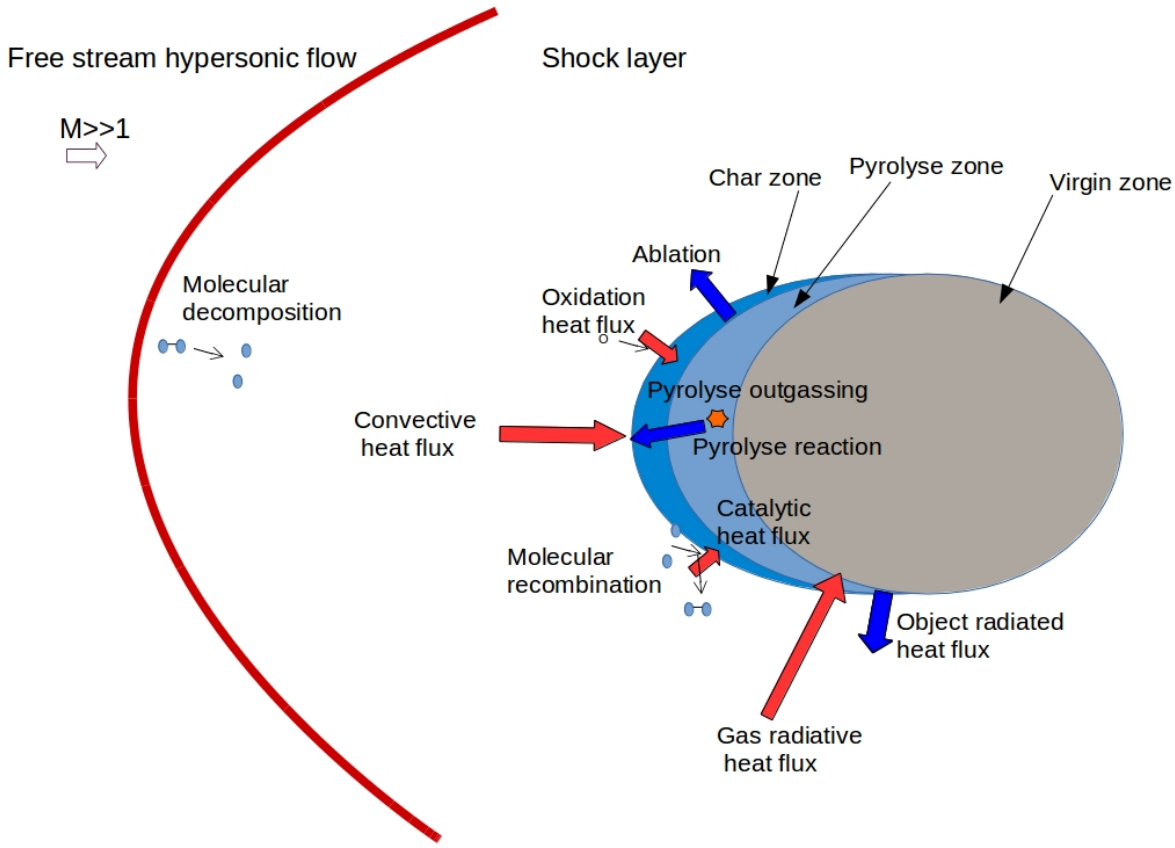
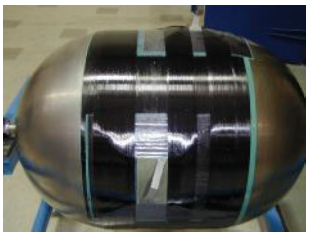


Figure 2.12: Schematic representation of interface phenomena between a high enthalpy flow and a composite material



(a) before reentry

(b) after reentry

Figure 2.13: Example of a cylinder tank made of CFRP composite material before and after rentry (source [8])

are caused by explosions due to battery burst, propellant leaks, and reactions in air or tank explosions. In the following, we give a description of upper stages and satellites. We then detail the breakup phenomena.

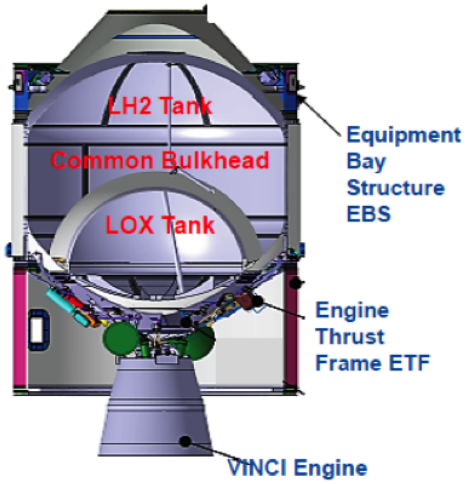


Figure 2.14: Ariane 5 ME upper stage (source : ESA)

Space object structural composition

An upper stage is the last stage in a staged rocket such as Ariane 5 and 6. It brings the payload as close as possible to its destination orbit. A simplified representation of the upper stage used in Ariane 5 ME is represented in Fig. 2.14. Most of the volume of the upper stage contains two tanks with the oxidizer and propellant. They are included in an aluminum structure assembled with titanium screws and flanked with thermal protection. The main components of the propulsion system are the turbopumps, the combustion chamber, the nozzle, and the pressure spheres. Those components are designed to withstand extreme temperature and pressure conditions and are mostly built in titanium or composite materials (CFRP). The electronic system is composed of batteries, power unit, and electronics which material composition is complex. The payload adapting system is responsible for carrying and releasing the payload. In the case of several satellites to send, the SYLDA (weighing around 500 kg) allows accommodating two satellites.

The shape and size of satellites vary according to their mission. Small satellites are typically cubes. The payload and components necessary for the functioning of the satellite (batteries, thermal control systems, data handling systems, etc.) are fixed to the walls. Additionally, solar panels are installed to ensure the power supply. The satellite is equipped with reaction wheels and a propulsion system (thrusters) to monitor the attitude and orbit. For larger satellites (GEO telecommunication satellites for instance) the structural integrity ensured by a central beam. In some cases, the beam is reinforced and also used as a tank. Most of the components are built in aluminum alloys although tanks are usually reinforced with CFRP or titanium.

Low energy breakups Low energy breakups are very likely to take place during the reentry of a space object. It is the separation of two components due to joint failure, welding failure or the fracture of an element. The mechanisms leading to breakup are extraordinarily diverse and still not well understood. Moreover, the observation of space object breakup is scarce and mostly qualitative insights from experts are available. Three major causes of low energy breakups can be identified: the thermal load, the aerodynamic load, and the inertial load. The three reasons are usually intricately coupled such as the modeling and simulation of breakup is exceptionally complex.

The thermal load corresponds to the energy brought by the flow that weakens the structure and makes it more likely to break or buckle due to the aerodynamic or inertial loads. The thermal

load may also cause material dilation in regions where the heat flux is high (near the stagnation point, edges or regions where the boundary layer thickness reduces). The material dilation adds extreme and localized constraints that may lead to a breakup in weak regions (wielding, joints). Finally, the thermal load may lead to the fusion of joints between two components and their separation.

The aerodynamic load corresponds to the aerodynamic forces that may lead to separation and buckling of the structure. Buckling is all the more likely to happen that the material temperature may be close to its fusion temperature. The aerodynamic forces are dominating on low ballistic coefficients such as solar panels, that are known to fragment early (around 100-90 km) during a reentry. The inertial load is due to the attitude motion of the object. In the case of an upper stage, the inertial load can be substantial due to the sloshing of the liquid fuel or oxidizer and the location of the center of mass at the rear.

High energy breakups High energy breakups come from internal vehicle explosions caused by battery failure, propellant leakage or tank explosion. Owing to the lack of experimental observations, the understanding of high energy breakups remains limited. On September 28^{th} 2008, the ATV 1 Jules Verne, responsible for the supply of the ISS performed a controlled, destructive reentry over the Pacific ocean that was thoroughly observed. A representation of the ATV is given in 2.15. It is composed of a docking system, pressurized module, control equipment and a propulsion system (gas mixing chamber, tanks, pressurization spheres, etc.). The objective of the observation campaign was to better understand the breakup of a reentering space object and the propagation of fragments. The first observed breakup happened at 74 km altitude [62] and probably originated from the propulsion system [10]. Before that, a flare was observed but is not identified in [10]. Half a minute after the main explosion, a second explosion is seen. During the same period, fragments are generated in the front face of the ATV, probably coming from the the protective shields and the docking adapter. Subsequent fragment separations are observed, but the breakup cause (low energy breakup or high energy breakup) is not elucidated in [10]. In Fig. 2.16, a picture from [10] of the ATV fragments is shown. The object fragmentation is usually a sequence of element separations and explosion over a minute time interval.

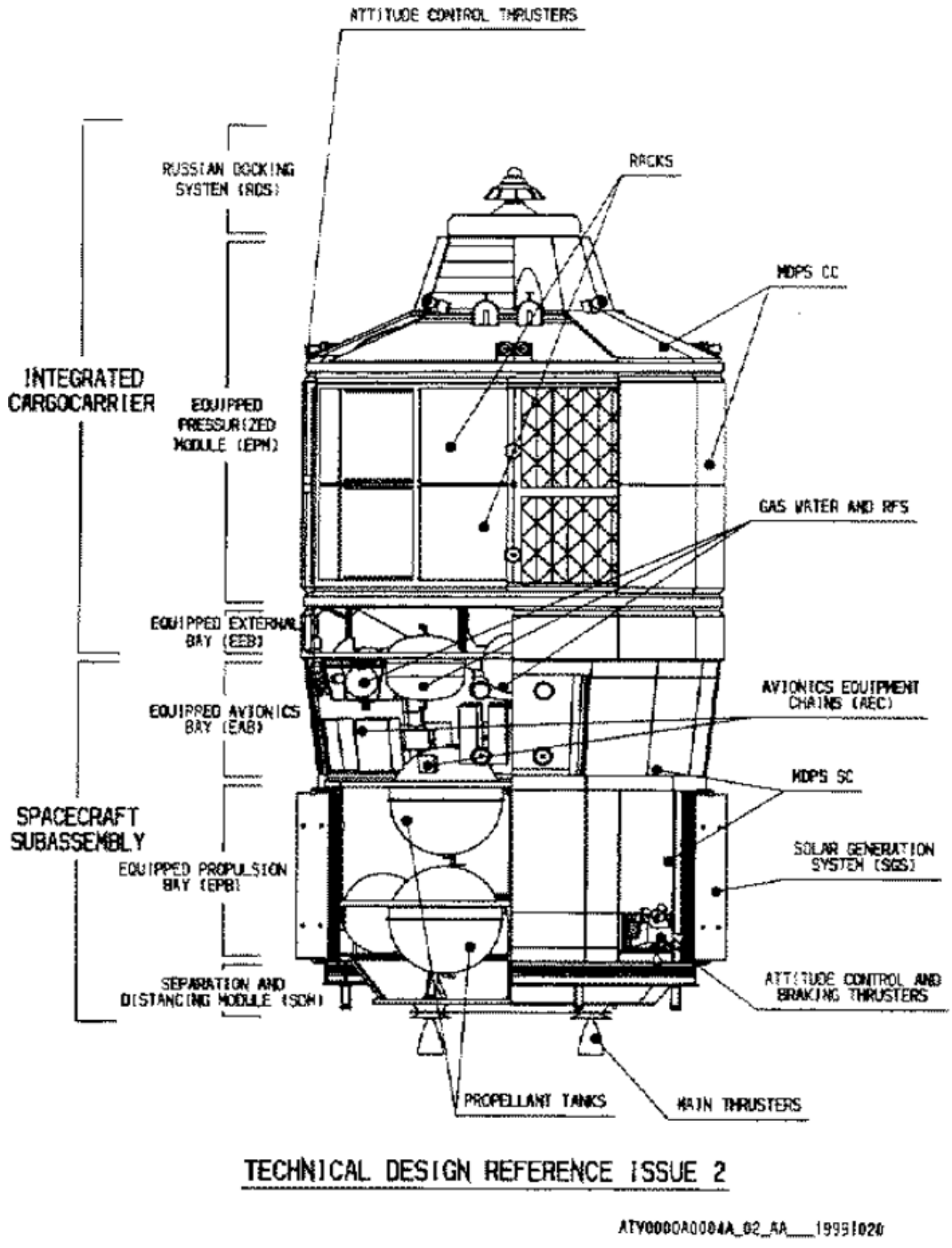


Figure 2.15: ATV main components (source [9])

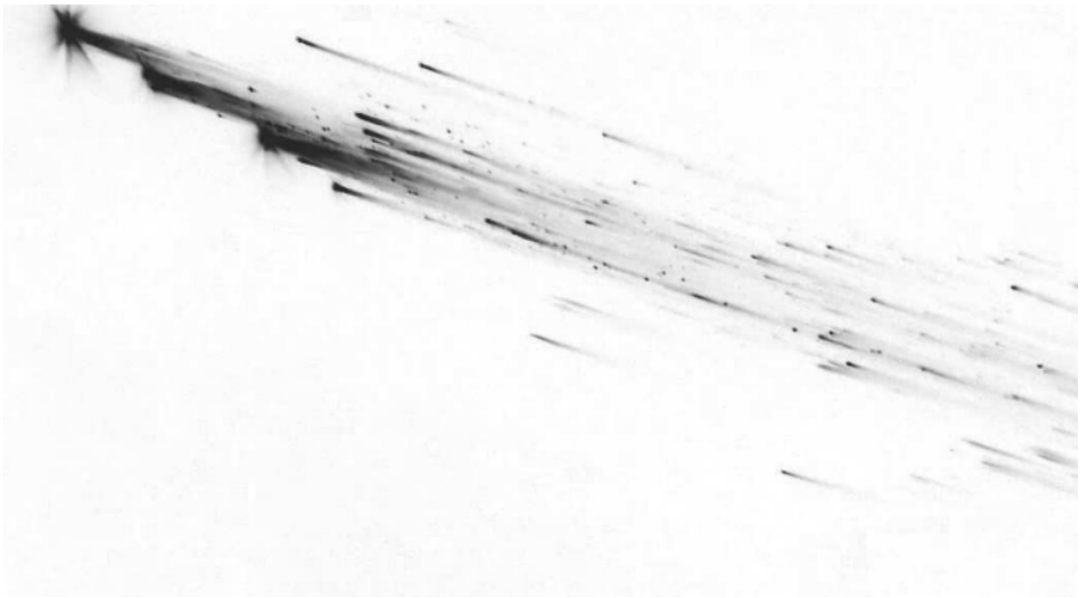


Figure 2.16: snapshot of the ATV Jules Verne reentry (source [10])

2.3 On-ground risk assessment

In the event of space debris reentry, the international guidelines define several risk metrics. The first one is the on-ground risk characterized by the probability of making at least one victim (P_v) and the expected number of victims E_v for a given reentry scenario. The model used to compute this risk is defined in [63] and has been implemented in [64] with minor differences. The second type of metrics are the Declared Re-entry Area (DRA) and the Safety Re-entry Area (SRA) defined in Chapter 1 and in [30]. Finally, the casualty area is the third type of metrics computed by space object reentry simulators.

2.3.1 Casualty area

The casualty area is the "equivalent impact area that lead a casualty if a person is struck by a piece of fragment" originating from the initial reentering space object [30]. Mathematically the casualty area for fragment k, $A_{c,k}$, is defined as :

$$A_{c,k} = (\sqrt{A_{f,k}} + \sqrt{A_h})^2 \tag{2.6}$$

where Af, k is the average projected area of the fragment surviving reentry and A_h is the cross-section of a standing human, taken equal to $0.36m^2$ in [30]. The projected area for simple shapes is given in [63] and reported in table 2.1. A graphical representation of the casualty area is given in Fig. 2.17. The total casualty area associated with a reentry event is defined as :

$$A_c = \sum_{k=1}^{n} A_{c,k} \tag{2.7}$$

with n the number of surviving fragments. The casualty area defines a level of hazard for each reentry, independently of the impact location and population density.

2.3.2 Expected number of casualty

The calculation of the expected number of casualty is proposed in [63] and implemented in [64]. The world population is modelled with a Poisson point process with density parameter $\lambda(x, y)$ where x and y are longitude and latitude. The density parameter $\lambda(x, y)$ is given by the world population model (gpw v4 [65]). For a given reentry, we denote $(x_1, y_1, ...x_n, y_n)$ the impact points of the surviving fragments 1 to n with casualty areas $A_{i,1}$,... $A_{c,n}$. We introduce the casualty factor c_k that is the probability that someone is killed knowing that it is hit by fragment k (i.e., it stands in the casualty area). According to [64], it depends on the kinetic energy of the object and the degree of protection of the population. In [63], it is recommended to consider unprotected population and therefore $c_k(\xi) = 1$ if the impact kinetic energy is higher than 15J and $c_k = 0$ otherwise. Note that the model ELECTRA uses a more refined model

Shape	Surface
Sphere radius R	πR^2
Box of dimensions $L > W > H$	LW
Plaque with $L > W$	LW
Cylinder with radius R and length L	max(πR^2,$2LR$)

Table 2.1: reference area for different debris shapes

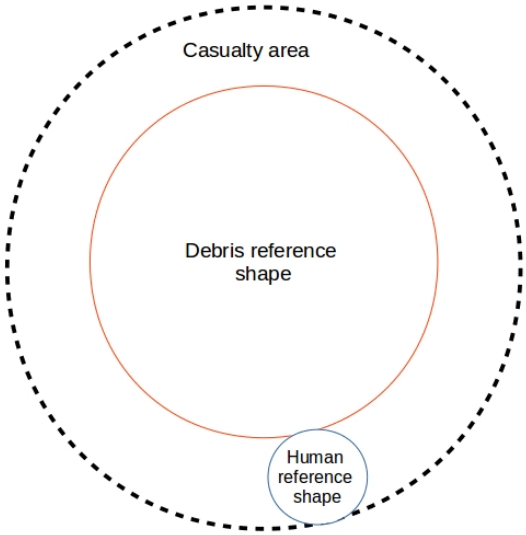


Figure 2.17: Casualty area definition

for c_k depending on the population location [64]. Casualty areas are small with respect to the population density resolution. Hence $\lambda(x, y)$ is constant over the casualty area. We also assume that the fragments have disjoint impact areas. Under these assumptions and the Poisson point process model, the expected number of victims is

$$E_v = \sum_{k=1}^{n} A_{c,k} c_k \lambda(x_k, y_k) \tag{2.8}$$

And the probability of having at least one victim is

$$P_v = 1 - \exp(-E_v) \tag{2.9}$$

Since the E_v is expected to be extremely small (on the order of 10^{-4}), we get that

$$P_v \simeq E_v \tag{2.10}$$

The casualty area metric is usually preferred to the expected number of victims as the risk estimation model is elementary and can solely be used as a certification tool. A significant advantage of the expected number of victims is that it takes into account the location of the impact. With the casualty area metric, a large object falling in the Pacific ocean is more problematic than a smaller object falling over a densely populated region. The casualty area is equivalent to computing the expected number of victims with a uniform density population.

2.3.3 Impact zones

The DRA and SRA defined in the guideline rely on a probability distribution of the impact location in terms of longitude and latitude from one fragment. Denoting $(X_1, \dots X_n)$ the vector

of joint distributions of the surviving fragment impact longitudes and latitudes. The X_i are by definition valued in $[-180, 180] \times [-90, 90]$. The DRA is the area of the set $\mathcal{A}_{DRA}$ such that

$$P(\cap_{i=1}^{n} X_i \in \mathcal{A}_{DRA}) \geq 1 - 10^{-4} \tag{2.11}$$

while the SRA is the area of the set $\mathcal{A}_{SRA}$ such that

$$P(\cap_{i=1}^{n} X_i \in \mathcal{A}_{SRA}) \geq 1 - 10^{-5}. \tag{2.12}$$

This definition is extremely fuzzy as there are a many possible $\mathcal{A}_{DRA}$ and $\mathcal{A}_{SRA}$ that satisfy Eqs. (2.11) or (2.12). In particular, if no constraint is added on the size of the impact zone it can be made extremely large by adding regions with zero impact probabilities. One possible restriction is to impose that the ground area covered by $\mathcal{A}_{DRA}$ and $\mathcal{A}_{SRA}$ be the smallest possible.

A rigorous solution to the impact zone determination problem has been proposed in [66] in the case of a unique fragment. In that case the number of candidates for the $\mathcal{A}_{DRA}$ and $\mathcal{A}_{SRA}$ can be reduced by considering minimum volume sets (MVS) only. This option is proposed by [66] to compute the fallback area of a launcher. In the unique fragment case, the joint distribution of impact longitudes and latitudes is denoted X. The quantity $\mathcal{A}_{DRA}$ is a minimum volume of level $1 - 10^{-4}$ if and only if

$$\begin{aligned} &\mathcal{A}_{DRA} \in \mathcal{P}([-180, 180] \times [-90, 90]) \text{ such as } P(X \in \mathcal{A}_{DRA}) \geq 1 - 10^{-4} \\ &\text{and } \forall B \in \mathcal{P}([-180, 180] \times [-90, 90]) \text{ such as } P(X \in B) \geq 1 - 10^{-4}, \quad L(\mathcal{A}_{DRA}) \leq L(B) \end{aligned} \tag{2.13}$$

where $\mathcal{P}([-180, 180] \times [-90, 90])$ is the set of partitions of $[-180, 180] \times [-90, 90]$, L is the Lebesgue measure (here the area of the sets). Hence, a MVS is a smallest set in terms of area such that $P(X \in \mathcal{A}_{DRA}) \geq 1 - 10^{-4}$. A priori, there are still a large number of possible MVS for a given probability level depending on the pdf of X. Moreover, finding a minimum volume set can be challenging. Fortunately, under regularity assumptions on the pdf, there exist a minimum volume set that is defined as a level of the pdf [67] . The idea is used in [66] where the authors compute the pdf in order to compute the level set. In their case, they consider extreme quantile and therefore need to have accurate estimates of the tails of the pdf. This approach has been implemented in [68, 50, 52] to compute fall back regions associated with a launcher failure.

In the general case where several fragment impact the Earth surface, the ideas presented in [66] do not apply. For space object reentry predictions with multiple fragments, state-of-the-art analysis use parametrized shapes for the impact zone such as rectangles or ellipsis [69] to propose an impact zone.

2.4 Conclusion

In this chapter, we describe the physics involved in the destructive re-entry of a human-made space object. The flow characteristics feature a diversity of phenomena, from the rarefied hypersonic flows to subsonic continuum flows. In the hypersonic flow conditions, the flow in the shock layer has converted its high kinetic energy into internal energy, leading to an increase of the flow enthalpy and even the molecular decomposition of air molecules. This high enthalpy induces intense convective flux and complex gas-surface interaction phenomena as molecular recombination, surface oxidation. The energy received by the object combined with the intense aerodynamic and inertial load leads to the fragmentation (low energy or high energy) of the object. A large panel of events can trigger breakup: joint or welding failure, structure buckling, epoxy glue melting or tank explosions. Depending on the type of breakups, the resulting fragments

will be extremely different in terms of mass and geometry. Hence, the breakup has a primary influence on the on-ground risk.

Lastly, we characterize the on-ground metrics proposed in the space agency guidelines. We identify three types of risk metrics: the expected number of victims, the casualty area and the impact zones that we define as the smallest zone such as the probability that a fragment falls outside of that zone is below a certain threshold.

Chapter 3

Reentry Simulations : Tools and Uncertainties

In this chapter, we illustrate the tools used to simulate the reentry of a space object and identify the uncertainties associated.
After reviewing the existing models available in the literature, we construct a reentry simulator constructed to peedict the controlled or uncontrolled reentry of a space object. This simulator can also be defined as a System of Solvers (SoS), which will be further discussed in Chapter 4. Despite using several heterogeneous solvers, we build a simulator requiring minimal user interactions, mostly focused on the definition of the initial/operating conditions and the scenario of interest. Minimizing the user interactions is necessary to perform Uncertainty Quantification efficiently, demanding otherwise the user to run one-by-one all the multiple simulations.
In this chapter, as the second contribution, we also identify the primary sources of uncertainty associated with object reentry modeling.

In the first section, we introduce the main two classes of reentry tools : the spacecraft-oriented and object-oriented models. In the second section, we present the reentry simulator constructed for this work. The level of fidelity is intermediate between object-oriented models and spacecraft-oriented models but with a probabilistic breakup model. Note that a higher fidelity model has also been developed in this work based on recent model improvements from AG. Since it is not employed in the uncertainty analysis of this work, we report it in Appendix B.

3.1 Review of object reentry simulators

There are two major classes of reentry simulators : the object-oriented and the spacecraft-oriented models.

In object-oriented tools, the spacecraft breakup is modeled with a parent-child approach. At a user-defined breakup altitude, the main (parent) object disappears and releases the child fragments. See Fig. 3.1. When a child object is released, it inherits its parent object state vector (position and velocity). A fragment is characterized by its geometry, dimensions, mass, attitude motion, and material. The shape and attitude motion of the object have to be selected from a limited choice of simple shapes (box,cylinder,cone, etc.). At each time step, the simulator computes the aerodynamic forces and heat flux and derives the object position, temperature and ablation rate. The simulator finally returns the impact locations of the surviving fragments. The functional representation of a generic object-oriented model is represented in figure 3.1. Examples of object-oriented reentry tools are DEBRISK and ELECTRA from CNES, DRAMA

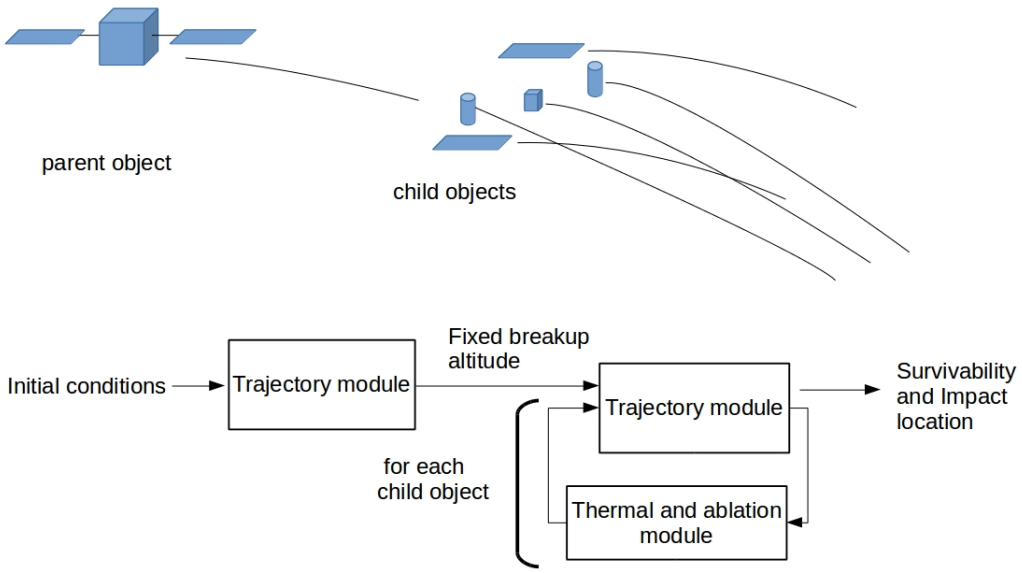


Figure 3.1: Schematic representation of object-oriented solvers.

from ESA, DAS and ORSAT from NASA, ORSAT-J from JAXA and SAPAR from South Corea space agency.

The spacecraft-oriented codes utilize more complex models compared to object-oriented codes. The significant difference is the spacecraft geometry representation. In spacecraft-oriented codes, the object is accurately represented in terms of geometry and mass distribution by a mesh and not with predefined shapes as in object-oriented models. Consequently, aerodynamic efforts and thermal loads can be computed more accurately. Moreover, spacecraft-oriented models feature six Degree of Freedom (DoF) trajectory solvers whereas object-oriented models do not solve for the object attitude motion. Finally, the breakup prediction is also significantly improved as it is not triggered at a given altitude but estimated using the thermal and structural responses of the space object to thermal and aerodynamic loads. Example of spacecraft-oriented simulators are PAMPERO from CNES, FAST/MUSIC from ONERA and SCARAB from HTG.

AG, in collaboration with CNES, developed a set of solvers for modeling the trajectory, aerodynamic loads and heat transfer that are used here to construct the reentry simulator.

In this section, we present more in-depth the object-oriented and spacecraft-oriented models and show how the solvers developed by AG compare to those models. In particular, we focus on the geometry modeling, the trajectory calculation, the aerothermal models, breakup models, and ablation models. Note that specific details for each solver are provided in Appendix A.

3.1.1 Geometry modeling

In object-oriented models, the child objects are necessarily modeled with predefined shapes (sphere, flat plate, cylinder or box) with a given motion attitude. For a given shape, dimension and attitude motion the aerodynamic coefficients are pre-computed and implemented in the solver. The Chinese space agency code, DRAPS, proposes a large number of objects and attitude

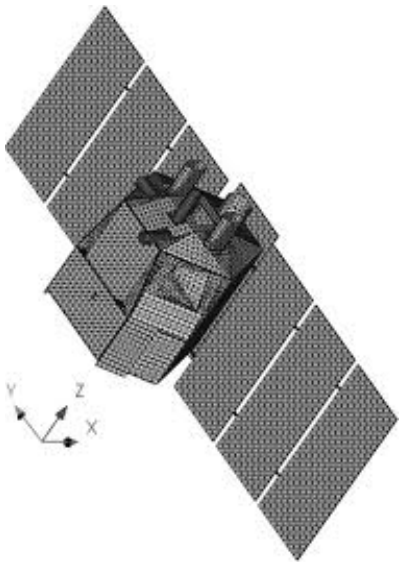


Figure 3.2: Representation Beppo-SAx (from [11])

motion (15 different geometries and 51 attitude motions) [6], but for other codes the choice usually is more limited.

On the other hand, spacecraft-oriented models do not use simplified shapes. Instead, the space object is represented by a mesh. In Fig. 3.2, a representation of the satellite BeppoSAX by SCARAB is shown. The geometry modeling tools also define connections between the components where the spacecraft may break up.

In the solvers developed by AG, the modeling of the object geometry is object-oriented for the aerothermal and thermal calculation but is spacecraft-oriented for the aerodynamic calculations. In other words, the object is approximated with simple shapes for computing the thermal response of the object but it is accurately modeled with a mesh for computing the aerodynamic efforts.

3.1.2 Trajectory modeling

The motion equations for a reentering space object are:

$$\frac{d^2\boldsymbol{X}}{dt^2} = A_e + A_c + A_g + \frac{\boldsymbol{F_{aero}}}{m}, \tag{3.1}$$

$$I \cdot \frac{d\boldsymbol{\Omega}}{dt} = \boldsymbol{M_{aero}} - \boldsymbol{\Omega} \wedge (I \cdot \boldsymbol{\Omega}), \tag{3.2}$$

where X is the position of the object in the Earth-centered inertial frame, A_g is the gravitational acceleration from the Earth, the Sun and the Moon, A_e the centrifugal acceleration and A_c the Coriolis acceleration. The term $\boldsymbol{F_{aero}}$ represents the aerodynamic forces. The quantity $\boldsymbol{\Omega}$ is the instantaneous rotation vector expressed in the engine inertial frame. The variable I is the inertia matrix associated with the object in the engine inertial frame. The term $\boldsymbol{M_{aero}}$ represents the moment induced by the aerodynamic forces

Most object-oriented models use a 3 Degree of Freedom (DoF) trajectory model. In other words, they only solve Eq. (3.1) and do not consider the object attitude motion. The spacecraft-oriented tools, on the other hand, solve for both Eqs. (3.1) and (3.2).

AG develops its own in-house trajectory solver (BL43) that may be used in a 3 or 6 DoF setting in which case Eqs. (3.1) and (3.2) are solved. Details are provided in Appendix A.

3.1.3 Case of controlled reentries : deorbiting modeling

Most reentry models are designed to simulate uncontrolled reentries. In general, the simulation starts at an altitude of 120 km. Before that, the trajectory is computed by a dedicated orbit propagator, such as STELA from CNES [70] for instance.

In this work, we are also interested in controlled reentries where the object is purposely ejected from its original orbit to permit fast and relatively precise reentry. To simulate the initial deorbiting manoeuver, AG developed the solver CARINE which is an orbit propagator that computes trajectories after a deorbiting manoeuver is performed. It takes as input the orbital elements of the initial orbit and the deorbiting characteristics: namely the deorbiting time t_{boost}, the retro boost amplitude A_{boost} and the boost orientation characterized by two angles, *i.e.* α_{boost} and β_{boost}. The new orbit characteristics are evaluated, and the orbit propagated until the object reaches 120 km of altitude. The model is less expensive than state-of-the-art orbit propagator such as STELA, but it is only valid for short time propagation, typically a deorbiting trajectory.

3.1.4 Aerodynamic modeling

To solve for Eqs. (3.1) and (3.2), we need to estimate $\boldsymbol{F_{aero}}$ and $\boldsymbol{M_{aero}}$ [1]. Most reentry tools (object-oriented and spacecraft-oriented) are based on the same set of assumptions. Three flow regimes are considered :

- In the free molecular flow, FAST and PAMPERO use Eq. 7.58 from the reference book [71] to get the local pressure and Eq. 7.61 in [71] to get the local shear stress. SCARAB uses similar results from the Schaaf-Chambre theory (see [72] ch 6). The local pressure and shear stress are then integrated over the object surface.

- In the continuum hypersonic regime, the local pressure is computed using the modified Newton law [4]. Besides, the shear tensor is set to zero which is reasonable for blunt bodies in hypersonic flows. The modified Newton law is valid for hypersonic flows and blunt bodies but becomes inaccurate for slender shapes and flow regimes below $M < 4$.

- For rarefied flows, the aerodynamic efforts are defined as a weighted sum of the aerodynamic efforts in the free molecular flow and the continuum regimes. The weights (also called bridging functions) are valued between 0 and 1 and depend on the Knudsen number.

While the same general aerodynamic models are used in the spacecraft-oriented and object-oriented approaches, they yield significantly different results on complex geometries. First, the calculation of the aerodynamic effort is more accurate in the case of the spacecraft-oriented approach (due to more refined geometry modeling). Second, some parameters such as the bridging functions differ in each code. In SCARAB and PAMPERO, for instance, the bridging functions are optimized according to experimental data [73, 74].

AG and CNES collaborated to develop the aerodynamic solver ARPEGE following a spacecraft-oriented approach. A surface mesh represents the object. ARPEGE uses the same set of assumptions as most solvers: Newton modified method for continuum hypersonic flows,

[1] Although $\boldsymbol{M_{aero}}$ is only evaluated in the case of spacecraft-oriented models

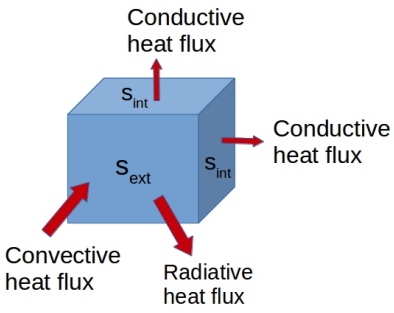


Figure 3.3: Representation of the heat flux applied to a cell.

bridging function in the rarefied flow and analytical expressions (eq. 7.88,7.89 from [55]) for the free molecular flow. More details are provided in Appendix A.

3.1.5 Thermal modeling

In this section, we discuss the differences in the object thermal response modeling for the object-oriented and the spacecraft-oriented approaches.

In the object-oriented models, only the parent object is exposed to the flow while the child objects are protected and are not included in the calculation of the thermal response of the model [75]. Moreover, a 0D thermal is usually used, such that the object temperature is assumed to be uniform. On the other hand, spacecraft-oriented tools use 3D finite element models to compute the thermal response of the object [6, 74], such that the spacecraft temperature field can be derived, and the conductive heat flux between parts of the object modeled. Hence, the energy conservation equations are not identical for the two approaches. For the object-oriented models, the energy conservation for the entire parent object is given by :

$$m_{ob} c_p \frac{dT_{ob}}{dt} = S q_{aero} \tag{3.3}$$

where T_{ob} is the object temperature, m_{ob} its mass, S its surface, c_p the material heat capacity and q_{aero} the aerothermal heatflux. For the space-oriented models, the energy conservation for a given cell is (see Fig. 3.3):

$$cp\rho v_c \frac{dT_c}{dt} = s_{ext} q_{aero} + s_{int} q_{cond} \tag{3.4}$$

where T_c is the cell temperature, v_c is the cell volume, s_{ext} the surface of the cell exposed to the flow and s_{int} the internal surface of the cell (not exposed to the flow). The quantity q_{cond} is the conductive heat flux model with Fourier's law [74, 6].

The aerothermal heat flux q_{aero} is not modeled identically in object-oriented and spacecraft-oriented models. In object-oriented models, it is the sum of the convective heat flux, the radiative heat flux and for some models, the oxidation heat flux [75] :

- The convective heat flux q_{conv} corresponds to the energy transfer induced by forced convection in the flow. The specific formulations of the convective heat flux are different for every software. In all cases, they rely on correlations and simplified heat flux formulations

from [76, 77, 78] in the continuum regime and the reference books [55] and [71] in the free molecular flow regime. In the rarefied flow regime, the heat fluxes are derived using bridging functions between the continuum flow and the free molecular flow expressions.

- The oxidation heat flux comes from the oxidation of the material in contact with dissociated oxygen atoms. The exothermic oxidation reactions generate additional energy absorbed by the material.

- The radiative heat flux corresponds to the re-radiation of the wall to the gas and the gas emission that occurs in the shock layer. In DEBRISK, the gas radiative heat flux is neglected while it is included in recent versions of ORSAT [79].

Spacecraft-oriented models also use analytical expressions that are usually improved with high fidelity calculations or high enthalpy experiments. The tool SCARAB for instance, also considers the heat flux from the pyrolysis reactions and outgassing cooling effects [60] for composite materials and FAST uses specific formulae to compute the gas radiative heat flux [6].

AG developed several models in collaboration with CNES to compute the aerothermal load and the object response. The tool ADRYANS V4 (described in details in Appendix A) follows an object-oriented approach: the object temperature is assumed to be uniform, and only the radiative and convective heat flux are modeled. Moreover, the spacecraft is approximated with simple shapes to compute the convective heat flux using simplified formulae. A parent-child approach is also selected, the object is modeled with a shell of mass m_{shell} containing all the fragments to be released at breakup. The shell protects the fagments from the flow. Note that AG also developed more advanced thermal models that are used in a second reentry simulator presented in the Appendix B.

3.1.6 Ablation modeling

As the object receives energy, it gets ablated by the flow. For metal alloys, it is generally assumed that the ablation process starts when the object surface reaches fusion temperature. The energy brought by the flow then serves to melt the material that is then swept by the flow.

For metal alloys, object-oriented and spacecraft-oriented models follow this approach but with a significant difference coming from the thermal model [75, 6, 80, 73]. For object-oriented models, the ablation starts when the entire object mass reaches fusion temperature (since a uniform temperature model is used). In the 3D thermal models implemented in spacecraft-oriented tools, the ablation process may start at the surface of the object before the internal cells reach fusion temperature as the conductive heat flux is generally lower than the convective flux. Consequently, the ablation process starts earlier with spacecraft-oriented models than with object-oriented models. Moreover, the ablation may be localized in spacecraft-oriented models whereas it takes place uniformly in object-oriented models.

On the other hand, for composite materials such as CFRPs, this approach leads to wrong survivability predictions [39]. In SCARAB, an improved model for composite materials is proposed [60]. Two additional heat flux sources are modeled in the case of composite materials: the pyrolysis reaction heat flux and the pyrolysis gas cooling [60]. The pyrolysis heat flux is a heat sink in the local energy balance equation owing to the endothermic pyrolysis reactions [60]. The gas cooling effect comes from the exhaust of pyrolysis gas that cools down the char region. In [60], the convective heat flux reduction induced by the pyrolysis outgassing [2] is also modeled. In

[2] Also called blocking effect

[39], the authors show the advantage of using this formulation compared to a metal ablation model for CFRP. This formulation, however, adds a significant number of parameters that have to be fitted. To the best of our knowledge, only SCARAB features a specific model to account for the ablation of composite materials.

The AG/CNES tool ADRYANS V4 uses a purely object-oriented approach, only valid for metal alloys.

3.1.7 Breakup modeling

To predict breakup, one needs to predict the object breakup chronology (there are usually several breakups during a reentry) and the properties of the generated fragments (shape, mass, material, temperature). The breakup is extremely simply modeled for object-oriented models: a set of precomputed fragments is usually released at a specific altitude chosen by the user [75]. Note that DRAPS uses thermal criteria to trigger breakup instead of a fixed altitude [81].

Spacecraft-oriented models, on the other hand, directly model the separation of the fragments from the original object. In SCARAB, the breakup is modeled using thermal and structural criteria. In earlier versions of SCARAB (1.5) [73], the separation of elements is triggered by the breakup of specified joints between two components. The structure breakup occurs when the stress computed at a joint reaches a certain threshold. The thermal breakup is detected when the joint between two parts melts.

Note that SCARAB has been equipped with a module to model tank failures during the ATV reentry study [9], but the model does not account for potential high energy breakups that can occur due to propellant leakage.

After each breakup, two or more fragments are generated. In general, one of them is the primary object while the others are small detached fragments that rarely experience further fragmentation [11]. After release, the pieces are usually modeled independently except for FAST that models the aerodynamic interactions between fragments [6].

In PAMPERO, the aerothermal response is coupled with the finite element code ASTER (developed by the company EDF) to perform the mechanical stress analysis [82]. To the best of our knowledge, no published work exists on the reentry of a complete space object with PAMPERO.

Additional breakup models have been proposed for high energy breakup in [61] using Bayesian failure network models. The proposed model is purely statistical and fitted on expert opinions and advanced opinion elicitation methods [61]. To the best of our knowledge, it is also the only model for high energy breakup during reentry. In [83], a probabilistic approach to fragment release is also proposed.

Since 2016, ArianeGroup engineers have been developing a breakup simulator. The objective is to compute the mechanical and thermal responses of an entire spacecraft to the hypersonic reentry flow along the trajectory to derive the breakup mechanisms. The solver is currently still being developed, and hence we propose here alternative models to simulate breakup. These models are presented in Section 3.2.2 and B.1.3.

3.1.8 Comparison results between object-oriented and spacecraft-oriented models

Object-oriented and spacecraft-oriented models have been compared in several studies [84, 85]. In [84], DAS (object-oriented) and SCARAB (spacecraft-oriented) are compared in their predictions of the casualty area, the residual mass, and the number of fragments reaching the ground.

The primary difference between DAS and SCARAB is the modeling of breakup and the generation of fragments. For DAS, a user-defined set of fragments is released at a user-defined breakup altitude. For SCARAB, the fragment characteristics and the release altitudes are computed by the model. In [84], the number and characteristics of the fragments for DAS are fixed, but the breakup altitude is varied from 58km to 98km. In all cases, the casualty area predicted by DAS is always lower than for SCARAB, even at low altitude (below 68 km) for which all the fragments survive. This study illustrates the consequences on the on-ground risk of the simplifying assumptions implemented object-oriented models.

In [86], the discrepancies in the impact location predictions between SESAM 3.1 (object-oriented) and SCARAB are investigated. As for DAS, the results in terms of ground impact risk, casualty area and impact location are significantly different, mostly due to the fixed breakup altitude, the initial fragment temperature and the fragment list that are chosen *a priori* by the user in SESAM 3.1.

In this section, we proposed an overview of the existing solutions to predict the reentry of a space object by establishing two main classes of models with different levels of fidelity. Not all reentry tools fall into one for those categories. As a example, we mention the software SAM [87, 88] developed by Fluid Gravity and Belstead as an interesting mix of both classes of models.

3.2 Construction of the full system of solvers for reentry

In this section, we describe the system of solvers for reentry prediction constructed during my PhD. The requirements for this tool are the following ones:

- The tool should provide an estimation of the ground risk, casualty area and in a probabilistic setting, the SRA and the DRA for a given reentry scenario.
- The tool should be fast and flexible. Hence the computational cost should remain within a few hours and require minimal user intervention.
- The tool should be robust to be used for Uncertainty Quantification (UQ) purposes. Then, the codes should run with any combination of parameters within the uncertainty ranges. We have performed rigorous testings of the in-house codes to check their stability to uncertain conditions.
- The tool should include a breakup model since the breakup solver from AG was not available.

We provide here a general description of the reentry prediction tool. Then, we detail the models that we developed, in particular, a breakup model and a risk estimation solver.

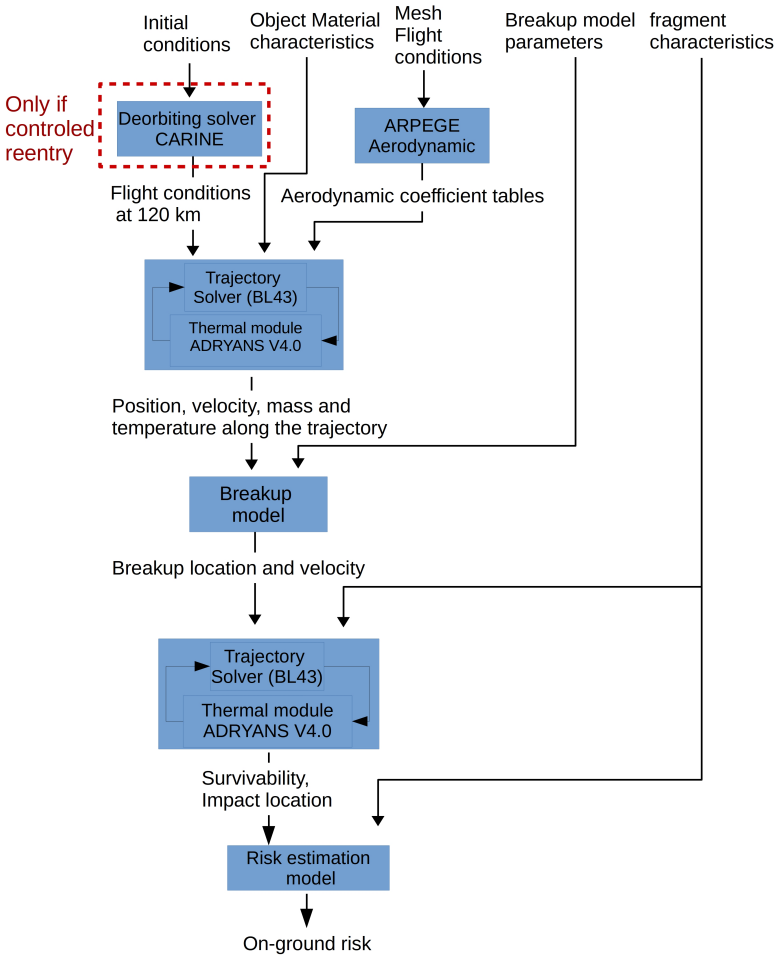


Figure 3.4: Functional representation of the medium fidelity simulator for controlled reentries

3.2.1 General description

Our simulator uses six or five solvers depending on whether the reentry is controlled or not. The details of the solvers developed by AG are given in the appendix A. A functional description of the simulator is given in Fig. 3.4 for the controlled and uncontrolled case. The two scenarios differ in their initial conditions. In the controlled case, the simulation starts at the deorbiting manoeuver while for the uncontrolled case, the reentry starts when the object crosses the arbitrary altitude of 120 km. Following figure 3.4 the simulator features the following steps:

1. At 120 km of altitude, the flight conditions of the object (velocity, position) are either given by the user (uncontrolled reentry) or computed by the deorbiting solver. In the case of controlled reentry, the deorbiting solver takes as inputs the initial conditions (initial orbit element, deorbiting maneuver characteristics) and returns the position and velocity of the object at 120 km altitude.

2. The aerodynamic coefficients associated with the object geometry are computed by the aerodynamic solver (ARPEGE). Tables are generated to avoid calling the aerodynamic solver

at every time step of the trajectory. They depend on the object attitude (orientation),the Mach number and the Knudsen number

3. The BL43 trajectory solver used in a 3 DoF setting is coupled with the thermal module ADRYANS V4 and the atmosphere model MSIS-2000 to compute the trajectory of the object (before any breakup). The model returns the object velocity, position, mass and temperature that are then necessary to compute the breakup conditions.

4. The breakup solver uses the temperature evolution along the trajectory and the trajectory to compute the breakup conditions and fragment release. As for the object-oriented models, the characteristics of the fragment to be released after the breakup, are precomputed by experts and high-fidelity simulations.

5. Given the fragment release conditions (velocity, position, temperature) and characteristics (material composition, shape, dimension, attitude motion) the fragment trajectory, temperature, and mass are computed. Two cases can be observed. Either the fragment disintegrates in the atmosphere, or it survives and impacts the ground. In the latter case, the solver returns the impact location (latitude and longitude), the residual mass, velocity, and casualty area.

6. For the surviving fragments, the risk estimation solver computes the human risk.

Note that we developed also a second reentry simulator. It features a more accurate breakup model and more advanced models for the heat flux calculation, the object thermal response and trajectory calculation. Since it has not been used with the uncertainty quantification tools, we have provided more details about this solver in Appendix B.

3.2.2 Breakup model

The trajectory solver BL43 coupled with the thermal module ADRYANS V4 computes the object position, velocity, mass and temperature ($\boldsymbol{X}(t)$,$\boldsymbol{V}(t)$,$m_{shell}(t)$,$T(t)$). The breakup module directly uses those quantities to compute the object flight conditions (position $\boldsymbol{X}_{breakup}$ and velocity $\boldsymbol{V}_{breakup}$) at breakup. This model follows a parent-child approach for object-oriented models. The parent object is the object shell with mass m_{shell} that contains and protects the fragments from the flow. The shell is approximated with a simple shape: cylinders for upper stages, boxes for satellites etc. At breakup, the parent object disappears and releases all the fragments at once or sequentially.

The particularity of this approach, compared to object-oriented models, is that the breakup solver is probabilistic and that the random breakup conditions depend on the flight history and not solely on *a priori* considerations. With this approach, we aim at including the uncertainties due to the extreme complexity of the physical phenomena.

The model constructs a random distribution of the breakup time. We define the breakup time distribution $\mathcal{T}_{frag}$ as:

$$\mathcal{T}_{frag} = \mathcal{U}(t_{init}, t_{end}), \tag{3.5}$$

where $\mathcal{U}$ denotes a uniform distribution and t_{init} and t_{end} are the lower and upper bounds of the time interval in which the breakup is bound to happen. The quantities t_{init} and t_{end} are computed using thermal criteria detailed in the following.

As the space object flies through denser and denser layers of the atmosphere, its temperature increases and its strength decreases until it reaches its fusion temperature. As the temperature

increases, the shell protecting the internal structure can fail before reaching fusion temperature due to the aerodynamic forces and the degraded structural performance [37]. Therefore, we assume that breakup can start as early as when the shell temperature reaches T_{frag} with $T_{frag} < T_{fus}$. Mathematically, t_{init} is defined as:

$$T(t_{init}) = T_{frag}, \tag{3.6}$$

where T_{frag} is a model parameter. It depends on the material and the reentry conditions.

When the temperature reaches fusion temperature, the material starts being ablated by the flow. Once most of the shell has been ablated, it is safe to say that the breakup has already occurred. Hence, we define t_{end}, the latest time for which breakup can occur, as the time where a certain mass proportion of the shell p_{abl} has been ablated. As for T_{frag}, p_{abl} is an uncalibrated model parameter. For a given p_{abl}, t_{end} is obtained by solving the following equation:

$$m_{shell}(t_{end}) = m_{shell}(0)p_{abl}, \tag{3.7}$$

where $m_{shell}(0)$ is the initial mass of the object shell.

Recalling that $\boldsymbol{X}(t)$ and $\boldsymbol{V}(t)$ are the position and velocity of the object at time t, the breakup position, defined as $\boldsymbol{X}_{breakup}$, is given by:

$$\boldsymbol{X}_{breakup} = \boldsymbol{X}(\mathcal{T}_{frag}). \tag{3.8}$$

Moreover, the breakup velocity, defined as $V_{breakup}$, is given by

$$\boldsymbol{V}_{breakup} = \boldsymbol{V}(\mathcal{T}_{frag}). \tag{3.9}$$

Note that after the breakup, the user-defined fragments are released with the initial conditions $(\boldsymbol{X}_{breakup}, \boldsymbol{V}_{breakup})$. Since $\mathcal{T}_{frag}$ is a random variable, $\boldsymbol{X}_{breakup}$ and $\boldsymbol{V}_{breakup}$ are random variables too. To make a prediction, one needs to generate a sample denoted t_{frag} from $\mathcal{T}_{frag}$ and the predicted breakup conditions are $\boldsymbol{X}(t_{frag})$ and $\boldsymbol{V}(t_{frag})$.

Discussion of the breakup distribution

The breakup distribution can be modified to include more *a priori* information about the breakup or even to model several breakups. For instance, instead of using a uniform distribution that represents a complete lack of knowledge about the breakup time between t_{init} and t_{end}, one could use more informative distributions such as the beta distribution with suited a, b parameters based on *a priori* object structure analysis. Moreover, for simplicity, the model assumes that there is only one breakup time although there are usually several breakups and fragment releases along the trajectory. This assumption, however, can be removed by considering distinct release times for each fragment. We define $\mathcal{T}^i_{frag} = \mathcal{U}(t_{init}, t_{end})$ the breakup time distribution associated with fragment i. The joint distribution of $(\mathcal{T}^i_{frag})$ can be chosen independent (each fragment is released independently from the others). On the other hand,if a chronology of the fragment release is available such that we know that

$$\mathcal{T}^1_{frag} < \mathcal{T}^2_{frag} < \dots \mathcal{T}^n_{frag}, \tag{3.10}$$

then a more informative breakup distribution can be used. Note however that the trajectory cannot be easily updated as fragments are released along the trajectory, although the object properties (mass, shape, inertia matrix etc.) are modified.

3.2.3 Risk estimation solver

A risk reentry solver is also implemented to complete the reentry simulator. It implements the equations described in Chapter 2 for computing the casualty area according to [30]. The casualty areas are computed using Table 2.1. The on-ground risk is computed using Eqs. (2.8), (2.9), and the NASA database GWP v4 [65] for the population density.

3.2.4 Assumptions of the proposed system of solvers for reentry

The major assumptions used here are:

- The trajectory is solved in a 3 DoF setting;
- The object temperature is uniform and heat flux calculations rely on simple correlations for simple shapes (box, cylinder);
- The breakup solver is based on an object-oriented approach but it introduces a stochastic feature that improves the robustness of the prediction. The disadvantage of this model is that it introduces two extra parameters T_{frag} and p_{abl} that cannot be easily estimated due to the lack of experimental results and the significant bias introduced by the thermal model.

As a comparison with existing reentry simulators, this simulator is close to object-oriented models but with the particularity of not having a fixed but a random breakup altitude that depends on the amount of energy received along the trajectory.

3.3 Implementation of the system of solvers

We tackled several challenges for the implementation of the system of solvers:

- The construction of an interface between codes that were written by different teams over long time periods using different programming languages and conventions;
- The automation and parallelization of the launches to make the simulator easy to use and automatic;
- The assessment of the accuracy of the solution.

We provide some details about the different actions in the following.

The python wrappers

Python wrappers are developed to build a unified framework in which the different solvers could be coupled. The objective was to provide the same input types and outputs independently on the structure of the solver. As the number of inputs could be significant, the dictionary structure was selected for the input and output communication. The wrapper task was to write the inputs in the input files, launch the executable and read the outputs in the output files. Improvements were added throughout this work to facilitate the use of the wrappers. Moreover, pre-processing and post-processing functions were implemented to allow the transparent use of data frames (from python package Pandas) in the case of large sample sizes. The used wrappers allowed to perform test efficiently and modify the system of solver while keeping a modular structure transparent to the user.

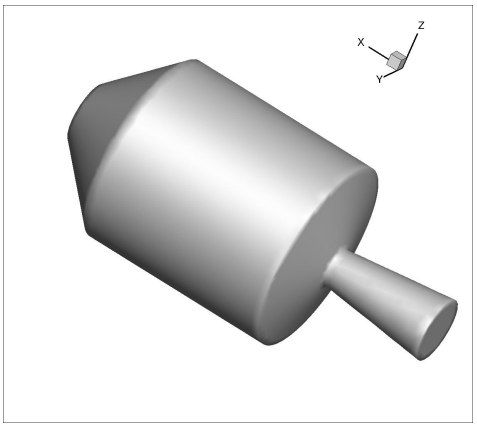


Figure 3.5: Representation of the upper stage for the medium fidelity model

Parallel launches

Based on the python framework built, the parallelization was first attempted using a distributed memory framework such as MPI. This option was not possible due to an incompatibility between the MPI-for-python module and the cluster architecture used for this work. Finally, the shared memory framework was selected although it could only run on a single node.

Convergence tests

For each solver, a certain number of tests had to be run to verify the accuracy of the results. The first class of tests assessed the numerical error due to the integration time step in the trajectory solvers coupled with the thermal module ADRYANS V4. Initially, the trajectory solver had an automatic time step adaptation method, but it did not consider the error in the thermal quantities such as temperature and mass loss. This induced large errors in the thermal quantities and sometimes even wrong predictions regarding the survivability of a fragment. For each piece, rigorous time step convergence tests were performed to verify the accuracy of the numerical methods.

The convergence of the mesh used to compute the aerodynamic coefficients with ARPEGE has also been tested. For example, in the case of an upper stage rocket represented in Fig. 3.5, three meshes with increasing resolution were tested. The calculated relative error between the coefficients with the three meshes was usually below 0.1 % and hence much lower than the uncertainties in the aerodynamic model. These tests were not only necessary for the application of UQ methods, but it also allowed detecting weaknesses in the codes and increasing the robustness of the implementation.

3.4 Reentry simulations in a deterministic scenario

We propose to test here the implementation of the simulator developed in this work and presented in the previous section. This test concerns the controlled reentry from a GTO orbit (see Table 3.1 for the orbit characteristics), which is the reference test-case used overall in this work. The space object is an upper stage represented in Fig. 3.5.

The upper stage is mostly composed of aluminium and weighs 7000 kg (see Table 3.1). The shell susceptible to be ablated weighs 3000 kg and is made of aluminium. For the breakup prediction, the parameter T_{frag} (see definition Eq. (3.6)) is set to 600K and p_{abl} (see definition

Table 3.1: Case description

Variable	Value
Internal mass	4000 kg
Shell mass	3000 kg
Material composing the shell	Aluminium
Dimensions	diameter 4.5 m , length (with nozzle) 7 m
Initial orbit apogee	35×10^3 [km]
Initial orbit perigee	254 [km]
Initial orbit inclination	10 [deg]
Initial orbit ascending node longitude	-135 [deg]
Initial orbit perigee argument	130 [deg]
Initial orbit true anomaly	43 [deg]
T_{frag}	600 [K]
p_{abl}	90 %

Eq. (3.7)) to 90 %. For simplicity, we consider a deterministic breakup time at $\frac{t_{end}-t_{init}}{2}$. Upon breakup, the precomputed fragments are released simultaneously. There are 29 types of fragments in total (see table 3.2), some are identical (for instance the pressure spheres) and hence fall exactly at the same location if released at the same time. The fragments are made of aluminium (75 %), stainless steel (5 %), titanium (10 %) and Inconel (10 %). There are globally 100 fragments released at breakup.

This reentry scenario is extensively studied in this work, in particular in Chapter 5 and 7 where uncertainties are included. For now we restrict ourselves to the deterministic simulation.

The model predicts a reentry at 9800 m/s at 120 km altitude. The fragmentation occurs 82.2 seconds after the beginning of the reentry around 64 km of altitude at 9400 m/s (almost Mach 32). The velocity and altitude evolution are represented in Figs. 3.6 and 3.7. Out of the 100 fragments only the four pressure sphere, the nozzle with the combustion chamber and a payload adaptor survive. The fragments fall into the Pacific ocean at several hundred-kilometer distances from each other, mostly due to the ballistic coefficient differences between the fragments (see Fig. 3.8). Overall, the total mass that reaches the ground is around 675 kg (less than 10% of the original mass). The impact velocities are subsonic (34 m/s for the spheres and up to 110 m/s for the nozzle and combustion chamber).

The simulator used in this deterministic setting offers little advantage compared to existing object-oriented reentry models. The breakup model is relevant only if used in a probabilistic setting that it would require a large number of fragment trajectory simulations. Moreover, the breakup model is not the only uncertainty in the model. As shown in the next section, there are several sources of uncertainties that should be identified and quantified.

Id	Number	Name
1	4	Payload adapter rings
2	1	Payload adapter
3	4	Cone
4	4	Equipments plate
5	3	Equipment bay structure 1
6	3	Equipment bay structure 2
7	4	Payload adapter Ring
8	4	Equipment bay structure 3
9	4	LH2 tank upper skirt 1
10	4	LH2 tank upper skirt 2
11	8	LH2 upper dome
12	4	Equipment bay structure 4
13	3	LH2 tank cylinder
14	4	LH2 tank lower skirt
15	4	LH2 tank ring
16	8	LH2 tank lower dome
17	3	Lox tank lower dome
18	3	Lox tank upper dome
19	2	Cold Gas Reacting System
20	8	Engine Thrust Frame 1
21	1	Engine Thrust Frame 2
22	1	Engine Thrust Frame 3
23	3	LH2 feedlines
24	3	LOX feedlines
25	4	Helium spheres
26	1	Combustion chamber
27	3	Power unit
28	1	Turbopump 1
29	1	Turbopump 2

Table 3.2: List of fragments

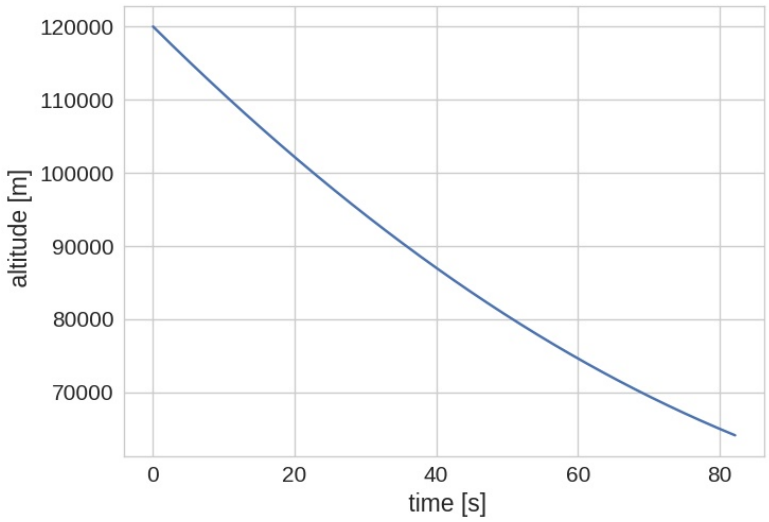


Figure 3.6: Time evolution of the upper stage altitude until breakup

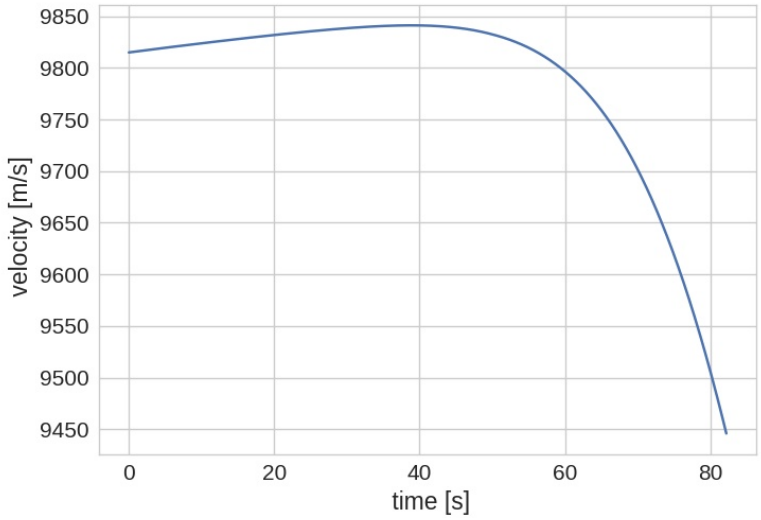


Figure 3.7: Time evolution of the upper stage velocity until breakup. The velocity increase is due to the low atmosphere drag above 100 and the gravity

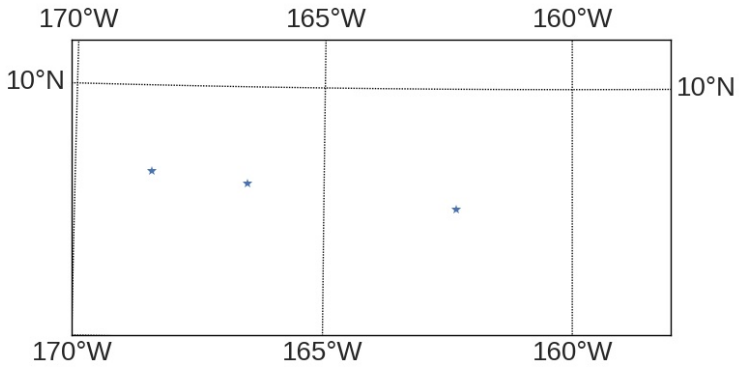


Figure 3.8: Impact location from (West to East) the pressure spheres (one impact point), the payload adaptor and the combustion chamber with nozzle

3.5 Uncertainty characterization

Despite a considerable effort from the community, even the most accurate models presented in this work use simplifying assumptions that reduce the reliability of their predictions. Those mismodeling errors induce uncertainties in the risk metrics (DRA,SRA, on-ground risk or casualty area). Moreover, the lack of knowledge of the reentry scenario (the position of the vehicle, the material composition or the mass) also affects the prediction reliability. We illustrate here two main sources of uncertainties.

3.5.1 Model errors and uncertainties

The solvers presented here feature different level of complexity depending on whether they are object-oriented or spacecraft-oriented solvers. In this section, we discuss the main model errors in both the spacecraft-oriented and object-oriented solvers. We also consider the differences between the object-oriented and spacecraft-oriented solvers and how the assumptions made by object-oriented solver significantly bias the risk estimation compared to spacecraft-oriented solvers.

Aerodynamic and aerothermal models

Most reentry solvers described in this work share similarities in their aerodynamic and aerothermal models. For the continuum flow, the modified Newton law is systematically used, even for complex, sharp shapes or supersonic flows with $M < 4$.

For simple shapes such as spheres, cylinders and flat plates [6, 12, 75], the agreement with high fidelity CFD models usually are very good for the drag coefficients and pressure distribution.

For sharp cones, FAST (ONERA) presents discrepancies with experimental results (around 10%) due to the contribution of shear forces [6] which are neglected in FAST (and most reentry simulators). Also, the sharp cone features a detached shock for large angles of attacks and potentially a subsonic region that is not modeled in the Newton law [6]. This example illustrates the error induced when applying the Newton law to sharp objects and arbitrary orientations.

Similarly, in the free molecular flow regime, the surface pressure and drag coefficient are well estimated for simple shapes [12, 80], but in the rarefied regime, the results are usually inferior. Recall that in the rarefied regime, all models use bridging functions to interpolate the results from the free molecular and continuum regime. The use of bridging functions simplifies the model but introduces additional errors.

Heat flux calculations are usually more challenging to evaluate. For spheres in the hypersonic continuum regime, the agreement is typically good [12] but large errors observed for plate plates (between 50 and 100 %) due to the mismodeling of the side faces of the plate. Also for the sphere, significant errors are observed in the rarefied flow regime (up to 42 % for PAMPERO and 30 % for SCARAB) [12] where bridging functions are used to estimate the heat flux.

In [80, 6], the spacecraft-oriented models are tested and compared with high fidelity simulations for more complex shapes. In the case of a tube, the drag coefficient is off by around 27 % compared to high fidelity calculations due to elliptical effects at the trailing edges of the front face and shock-shock interactions in the inner face of the tube. Those effects also induce a significant error in the heat flux (around 50 %). Similar discrepancies are observed with a hollow sphere in the hypersonic continuum regime as the Newton modified law is valid for concave objects. As a consequence, the front face pressure field is entirely off with the Newton method.

Figure 3.9: Photography of the Atmospheric Reentry Vehicle at ESA ESTEC

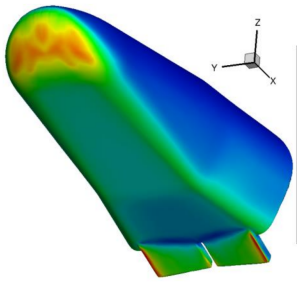


Figure 3.10: Representation of the PRE-X (from [12])

Similar tests were performed on the Atmospheric Reentry Demonstrator (ARD) for which high fidelity calculations and experimental measurements are available. The ARD is a reentry capsule represented in Fig. 3.9. The ARD was tested in the continuum regime (altitude 65 km) at Mach 24. The spacecraft-oriented codes are in good agreement with the experimental and high fidelity results with a maximum error of 5 % at the stagnation pressure coefficient and less than 10% for the stagnation point heat flux. These good results come from the simple geometry and convex shape of the front face of the ARD.

The second full-scale test is the PRE-X represented in Fig. 3.10. In the stagnation point area and the underneath flat surface, the PAMPERO and FAST are in good agreement with the CFD calculations. The main differences are observed near the flats fixed at a deflection angle of 15 degrees where the pressures increases and the heat flux reduces due to a flow compression. As for simple objects, the spacecraft-oriented model performance heavily depends on the local geometry of the spacecraft. If the object is convex with smooth angles, the results are satisfying.

From the test cases presented significant errors in the aerodynamic and aerothermal models come from:

- The bridging functions used in the rarefied flow regime (especially for the heat flux);
- The sharp edge objects for which the modified Newton law is not valid;
- The trailing edge elliptic phenomena;
- The treatment of hidden surfaces with Newton method;

- Application of the Newton law on concave surfaces ;
- Subsonic flow in recirculation zones.

Object modeling

In state-of-the-art spacecraft-oriented model, a mesh with a given mass distribution represents the spacecraft. While the error in the mass of the object can be made extremely small [11], the inertia matrix and the center of mass location may be wrong. All the details of the spacecraft electrical components cannot be modeled accurately. In [11], the battery weight is compensated by a thicker wall to obtain the right total mass, but the difference in mass distribution may induce significant differences in the inertia moments and the object attitude motion. Similarly, the liquids in the tanks are modeled as solids such as sloshing effects are neglected. All these approximations modify the object attitude motion, trajectory and received heat flux.

Breakup modeling

Although breakup models in spacecraft-oriented software are more advanced than object-oriented models, they are still subject to mismodeling errors. Regarding low energy breakups, the use of fully coupled 3D finite element solvers for the thermal and mechanical response of the object to aerodynamic and thermal load is costly in terms of computational cost and engineer time. In practice, SCARAB often models breakup solely using a thermal model, by triggering fragmentation when the joint between two components melts. This approach neglects breakups entailed by aerodynamic loads. Besides, the criterion is also questionable since the joint may break before it reaches fusion temperature such that SCARAB may underestimate the breakup altitude in these cases. Experimental studies show that metals are significantly weakened when their temperature increases such that they are more likely to break before melting [89, 90]. More advanced mechanical models could be used as in PAMPERO, but they are costly to use, and as for lower fidelity models they cannot be compared to experimental results that are extremely rare and complex to interpret [10]. High energy breakups are usually neglected in space object reentry simulations. As a consequence, the shape of the fragments and the altitudes at which they are released is subject to uncertainties.

Material modeling, surface effects

Hypersonic flows are characterized by reacting boundary layers, but this aspect is partially considered in reentry tools. The molecular recombination of atoms at the surface of the object is not modeled although it contributes significantly to the convective heat flux. The oxidation reactions are also only modeled as a contribution to the heat flux, but the alteration of the surface properties are not taken into account. Numerous studies have shown that the material properties are heavily modified when the object is oxidized [91, 59, 8, 6]. The emissivity of metal such as titanium tends to increase when oxidized [59, 6]. The alumina created by oxidation of aluminum has a fusion temperature around 2300 K whereas aluminum is approximately 800 K. As a consequence the surface temperature of an aluminum object covered with alumina can reach 1100 K and radiate heat efficiently [8]. Consequently, the survivability of this object increases compared to models predictions [8]. For titanium, different observations are formulated in [6]. According to the author, the oxide layer contributes to a faster ablation of the material.

For the composite materials, SCARAB features a model for the ablation of composites, but it requires a large number of ill-calibrated parameters (erosion rate, pyrolysis gas heat capacity) and the ablative process of composite materials is still not well understood.

Human risk estimation model

The distribution of human population used in ELECTRA, for instance, is based on the GWP [65] maps established by NASA and has a limited resolution as it assumes a zero population density in the ocean neglecting commercial ship roads. This fact may be an issue as specific zones in the Pacific ocean are targeted for the controlled reentry of space objects as they feature a minimum on-ground risk according to the GWP maps. Finally, some predictions have to be performed for reentries in 20 years from now, the extrapolation of current data to future population density is undoubtedly affected by uncertainties.

Object-oriented model errors

Based on the discussion in section 3.1.8 and the work from [84, 86], the main modeling errors from object-oriented models are :

- The single breakup altitude: the results from SCARAB have shown that the breakup is not a single event but that fragments are released sequentially over an altitude range around 95 km for the solar panels and down from 45 to 80 km for the main body [86, 84]. In the case of a satellite reentry, ref. [86] shows that the fragment release is relatively uniform except for the solar panels. Hence the assumption that all the object should be released at once is far from higher fidelity simulations and observations (ARD Jules Verne reentry).

- The fixed breakup altitude: in object-oriented models, the breakup altitude is fixed and chosen independently of the reentry trajectory. Besides, it seems that the canonical altitude of 78 km is too high to provide a reliable estimate of the casualty area [84].

- The fragment temperature at breakup: the typical fragment release temperature is usually set to 300 K for conservative estimates assuming no heat transfer between the parent body and the child components. As shown in [86], the fragment temperature is certainly much higher due to internal radiative and conductive heat flux.

- The precomputed list of fragments: it is expected that the shapes of the fragments depend on the aerodynamic and thermal loads received by the object before fragmentation. Depending on the reentry conditions, the fragment geometries change.

- The simple shape models: the object-oriented codes use predefined simple shapes with given motion attitude. When comparing ORSAT and SCARAB on simple shapes, similar results were found [85] because the aerodynamic and aerothermal models are similar. However, in a more realistic case where a complete spacecraft with an arbitrary shape is considered, significant differences between spacecraft-oriented and object-oriented solvers are observed in terms of impact location [86]. This remark applies to satellites and upper stages. The influence of the heat flux has not been quantified in [86, 84] but it is reasonable to assume that the simple shape models also significantly biases the heat flux predictions.

3.5.2 Uncertainties on model parameters and operating conditions

The second source of uncertainties comes from the particular object situation and is independent of the model errors and uncertainties presented in the previous paragraphs. This second source of uncertainties concerns the model parameters and operating conditions.

Atmosphere variations

The turbulent atmosphere features abrupt density changes in high altitude (above 100 km) and unpredictable winds in low altitudes where the fragments fall at subsonic speed. In the most advanced atmosphere model used in space object reentry simulators (MSIS-2000), the average atmosphere characteristics (density pressure temperature) are computed for a given time, position, solar activity, and Earth geomagnetic activity that may not be known for a given reentry scenario.

Object position and attitude motion

In general, the exact attitude motion and position of a reentering space object is not entirely known especially for uncontrolled reentries. In the latter case, even the time of reentry is estimated with uncertainties beyond an orbit. The orbit decay usually takes up to dozens of years before reentry. Meanwhile, the object position determination typically relies on noisy on-ground observations [92, 93] and complex long term orbit propagation simulations subject to uncertainties [25].

For controlled destructive reentries, the uncertainties are usually much lower depending on the knowledge of the original orbit and the accuracy of the deorbiting maneuver.

Material characteristics

A broad range of material characteristics is used in both spacecraft-oriented and object-oriented models. The material characteristics are measured experimentally, and therefore their value is naturally subject to uncertainty. In the case of the fusion temperature, the thermal conductivity, the thermal dilatation coefficient and the emissivity, the measurement procedure is standard, and the results are accurate. On the other hand, the CFRP recession rate or the wall catalytic effects on molecular recombination are much more complicated to reconstruct [94, 95]. The material structural properties (buckling limit, the transition from elasticity to plasticity) are also rarely known at extreme temperatures typically observed during a reentry.

Partial knowledge of the spacecraft

In the case of an uncontrolled reentry, the exact material composition, mass and geometry of the object may be unknown. Uncontrolled reentries concern space objects that have been decaying for years and therefore, they may have gone through collisions, in-orbit explosions, and in-orbit breakup that modify the object geometry. Moreover, if the design is protected, the internal structure is also unknown as well as the material used in the construction. The state of the tanks is also subject to uncertainties as the quantity of liquid propellant inside may be unknown. This uncertainty affects the center of mass position and total mass.

3.6 Conclusion

In this chapter, we construct a system of solvers for the reentry using solvers developed by ArianeGroup and CNES. It features a probabilistic breakup model accounting for the extreme complexity of the breakups that could not be resolved with the resources and models available. The system of solvers is implemented in a python framework that offers great modularity and flexibility.

Compared to the existing models presented in this chapter, the simulator uses object-oriented type models but includes a more advanced breakup model that offers improved robustness

compared to models with fixed breakup altitudes. The drawback is that the breakup model should be coupled with a large number of Monte Carlo runs to be reliable and it would be expensive without dedicated UP methods.

As any reentry simulation tool, the reentry simulator presented in this chapter suffers from uncertainties and model errors. We identified the primary sources of uncertainties found in both object-oriented and spacecraft-oriented models. The uncertainties come from the simplified heat flux and aerodynamic models, the unknown initial conditions or the atmosphere fluctuations for instance.

Uncertainty analysis has been performed in the literature, but they are based on simple sensitivity analysis methods or a limited number of Monte Carlo samples. It is, therefore, crucial to find efficient ways to include uncertainties to our model and use the probabilistic breakup model. In the next chapter dedicated methods for propagating uncertainties in directed systems of solvers are proposed.

Chapter 4

Uncertainty Propagation Framework for Systems of Solvers

In the first part of this chapter, we present a traditional mathematical framework used for Uncertainty Quantification and standard tools associated. In the second part, we propose an original framework for uncertainty propagation in directed systems of solvers as the one constructed in Chapter 3. The contributions of this chapter are

- *A novel framework to construct surrogate models for a system of solvers using Gaussian Processes.*
- *Dedicated design of experiment techniques for systems of solvers including active learning strategies*

The tests run at the end of the chapter highlight the advantage of the approach compared to state of the art methods.

4.1 Introduction to uncertainty quantification

In this section, we introduce a few general concepts in uncertainty quantification related to our problem : how to model uncertainty? what is uncertainty propagation and why is it computationally expensive?

4.1.1 Uncertainty modeling

In space object reentry predictions as for many engineering problems, the *Quantities of Interest* QoI are defined as the physical quantities we want to estimate to address an engineering problem. The QoI are obtained through the evaluation of numerical models that depend on a set of inputs. In an uncertainty quantification problem, the computation of the QoI can be affected by uncertainties coming from three different sources: noisy experimental data, unknown model parameters, and model errors. Model errors come either from the inaccurate resolution of the equations associated with the model (numerical errors) or the inadequacy of the model with the physics. In the case of space object reentry, examples of noisy data can be for instance the space object observations before its reentry. Examples of unknown input parameters are the atmosphere model parameters such as the solar flux. The model numerical errors have been assessed in Chapter 3 during the verification step and include mesh approximation errors or time integration errors. Examples of numerical inadequacies are for instance the use of the Newton

law for subsonic flows as discussed in Chapter 3.

In this work, the uncertainties are modeled as random variables. This approach is classical for uncertainty quantification problems although it may not necessarily be adapted for all kind of uncertainties. Uncertainties are canonically classified into two groups [96] : the aleatory and the epistemic uncertainties. The aleatory uncertainties are naturally modeled as random variables since they measure the intrinsic variability of a quantity that cannot be reduced or controlled. In our case, they could be the random variations of the atmosphere density or the solar flux induced by changing solar activity. Epistemic (or reducible) uncertainties are related to a lack of knowledge. They arise from assumptions in the models, unknown initial conditions or uncalibrated model parameters. The frontier between aleatory and epistemic uncertainties is sometimes fuzzy and subjective [97]. Modeling epistemic uncertainties as random variables is more controversial. Alternative approaches propose to model epistemic uncertainties as intervals, fuzzy sets [98] or imprecise probabilities [99, 100]. In this work, we follow a Bayesian interpretation of probability as a degree of belief. This interpretation proposed by De Finetti is extensively presented in [101]. Hence in this work, probability distributions are used to model both aleatory and epistemic uncertainties.

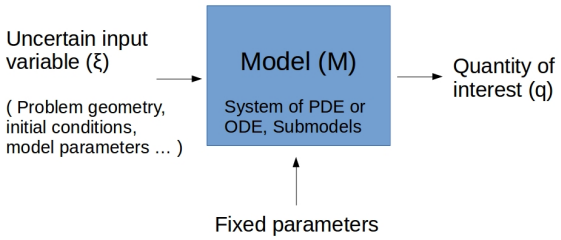


Figure 4.1: Uncertainty propagation framework to derive the QoI distribution from the uncertain model inputs

We will consider a numerical model M that generates a mapping between the uncertain variables $\boldsymbol{\xi}$ and the QoI q (see fig.4.1). The $\boldsymbol{\xi}$ are defined on the probability triplet $(\Theta, \mathcal{F}, \nu)$ where Θ is the set of outcomes , $\mathcal{F}$ is the sigma field associated and ν is the probability measure induced by the random vector $\boldsymbol{\xi}$. We define the probability density associated with $\boldsymbol{\xi}$, p_ξ. The random vector $\boldsymbol{\xi}$ is valued in the input space Ω of dimension d. In this work, Ω is a subset of $\mathbb{R}^d$. The quantity q is valued in $\mathbb{R}$. We denote $f : \Omega \mapsto \mathbb{R}$ the mapping representing M, between the uncertain inputs ξ and the QoI q. We further assume that $f(\xi)$ has a finite second moment, hence

$$\int_\Omega f(\xi)^2 p(\xi) d\xi < \infty \tag{4.1}$$

and f lives in the L_2 Space $L_2(\Theta, \mathcal{F}, \nu)$ equipped with the inner product generated by the expectation operator :

$$\int_\Omega f_1(\xi) f_2(\xi) p(\xi) d\xi. \tag{4.2}$$

The objectives of Uncertainty Propagation (UP) are to derive the distribution of q through its moments, to compute probabilities associated with q (for instance in reliability problems) and to perform sensitivity analysis.

4.1.2 Uncertainty propagation methods

There are two broad classes of UP methods: the *intrusive* methods and the *non-intrusive* ones. The non-intrusive methods consider the model M as a black box and aim at propagating the input uncertainties without any modification of the numerical methods implemented in M. Consequently, non-intrusive methods are flexible and can be used on any black box type model. On the contrary, intrusive methods require a reformulation of the model equations. Non-intrusive methods are generally selected in industrial applications for their flexibility and the possibility to run the multiple calls of M in parallel. The main drawback of most non-intrusive methods is that the number of calls to M scales exponentially with the number of independent random inputs [102]. If M is an expensive industrial solver, this undesirable property referred to as the *curse of dimensionality* [103] may become problematic. Most works realized in non-intrusive UP aims at mitigating the curse of dimensionality.

The first class of non-intrusive methods to propagate uncertainties are sampling methods. The sampling approaches are straightforward to implement, robust to non-smooth models and their convergence rate is usually independent of the input dimension. The major drawback of standard Monte Carlo approaches is that the estimator error typically scales as $1/\sqrt{n_{eval}}$ where n_{eval} is the number of solver evaluations. In industrial cases, where the resolution of M may be costly, higher convergence rates are needed. Improved sampling strategies such as Quasi Monte Carlo QMC [104] and Latin Hypercube Sampling LHS [105] have been developed to accelerate the convergence rate. The general idea behind improved sampling methods is to improve the coverage of the input space and minimize the discrepancy. Asymptotically, LHS achieves the same error convergence rate as MC but with a smaller constant for additive problems [106] while QMC error converges with an improved rate of $\log(n_{eval})^d/n_{eval}$ [102] depending on the input dimension d.

The second class of methods is the spectral approaches. Although spectral methods can be intrusive or non-intrusive, in this review, we focus on non-intrusive methods. A thorough review of intrusive methods can be found in [102]. A popular non-intrusive spectral approach is the Polynomial Chaos Expansion PCE for independent inputs. This approach aims at approximating the QoI with a (finite) expansion of a polynomial basis orthogonal with respect to the input probability measure [102, 107]. The idea was first introduced by [108] for Gaussian input distributions and extended to a large number of common probability laws by [107] such that generalized PCE can be constructed on arbitrary independent input distributions using the Askey scheme [107] or numerical methods such Gram-Schmidt algorithm or the Stieltjes procedure [109]. A review of methods to compute the coefficients of this expansion can be found in [102]. In particular collocation methods, non-intrusive spectral projections [110, 111], and regression methods [112] have been proposed. The PCE methods perform well for smooth problems and present a spectral error convergence rate with respect to the polynomial degree p in the L2 norm [113]. The convergence rate depends strongly on the model smoothness, and for non-smooth problems (discontinuities), Gibbs phenomenon may appear [113]. As for most non-intrusive methods, the PCE suffer from the curse of dimensionality in the estimation of the coefficients and the number of coefficients to consider.

The third class of non-intrusive uncertainty propagation approaches are based on surrogate

models. As sampling methods are too costly to be used with expensive physical models, one approach is first to build a cheap approximation (also called surrogate model or metamodel) of the model M represented by the mapping f and then to use standard Monte Carlo estimators to derive the statistics of interest of q. Hence, the objective is to build a mapping $\hat{f} : \Omega \mapsto \mathbb{R}$ such that $\hat{f}$ is a good approximation of f. Qualifying the performance of $\hat{f}$ as a surrogate model depends on the statistic of q one wants to reconstruct. One classical approach is to look at the distance between $\hat{f}$ and f in the L_2 norm induced by the inner product in Eq. (4.2). A good surrogate model should be constructed upon on a limited number of solver evaluations called training points and should accurately emulate the model mapping f. The generalized regressions are standard surrogate modeling constructions that minimize the squared error between a linear combination of a given functional basis and f at the training points [114]. Examples of functional bases are polynomials or radial basis functions [114]. For high dimensional cases, sparse methods such as lasso or ridge regression allow reducing the number of coefficients in the regression using regularization [114]. Sparse regression methods have been successfully applied to uncertainty propagation in [112, 115]. Alternatively, Neural Networks [116] are also gaining interest from the UQ community [117, 118]. Non-parametric approaches such as Kriging [119] also called Gaussian Processes [120] have also become popular for uncertainty quantification problems [121, 122, 123] and the contributions on this work rely on this type of constructions. A more in-depth presentation of Gaussian Processes can be found in paragraph 4.1.4. The PCE methods mentioned in the previous paragraph can also be considered as surrogate model approaches as the constructed polynomial is an approximation of the mapping f. As for most non-intrusive methods, the performance of the surrogate model is highly dependant on the input dimensionality. For this reason dimensionality reduction methods have been proposed in [124, 125, 126].

The performance of the surrogate model is highly dependent on the quality of the training set. The design of experiments is a complex task with intensive ongoing research aiming at selecting the optimal training samples to provide the best approximation of f. In paragraph 4.1.5, we review classical approaches to obtain good designs of experiments.

4.1.3 Sensitivity analysis

Sensitivity analysis SA aims at estimating the influence of the uncertain input parameters on the QoI. It can be useful to identify dominating uncertainties and discard non influential uncertainties. In [127], it is defined as "The study of how uncertainty in the output of a model (numerical or otherwise) can be apportioned to different sources of uncertainty in the model input". There are two categories of SA : local SA and global SA. Local SA studies local variations of the model output q with respect to the model inputs at fixed input locations. A classical local sensitivity index is for instance the derivative of the model when defined. This approach however is often ill adapted for UQ problems as the input quantities are not fixed but uncertain [127].
Global SA, on the other hand, quantifies the global influence of an input over its distribution and the possible interactions between inputs. The screening approaches are classical engineering methods for global SA. They study the variation of the output with respect to a discretization of the input space. This approach is for instance used in the uncertainty study presented in [38]. The advantage of screening techniques is their low computational cost even with large number of uncertain parameters. The main drawback is that they only provide qualitative results and often neglect interactions. A review of classical sensitivity analysis methods is presented in [128]. In this work, we use the Sobol indices [129, 127] based on the variance decomposition of the output variance into each input contribution. They are defined as :

$$S_i = \frac{\mathbb{V}\left[\mathbb{E}\left[q \mid \xi_i\right]\right]}{\mathbb{V}\left[q\right]} \tag{4.3}$$

for the first order indices. The first order Sobol indices V_i accounts for the contribution of input ξ_i to the variance of the output neglecting its interactions with other variables. The first order Sobol indices do not account for the possible input interactions leading to extreme values of q for instance. To account for interactions, second order sensitivity indices are defined as :

$$S_{i,j} = \frac{\mathbb{V}\left[\mathbb{E}\left[q \mid \xi_i, \xi_j\right]\right] - \mathbb{V}\left[\mathbb{E}\left[q \mid \xi_i\right]\right] - \mathbb{V}\left[\mathbb{E}\left[q \mid \xi_j\right]\right]}{\mathbb{V}\left[q\right]}. \tag{4.4}$$

Similarly, higher order interaction indices can be constructed. They quantify the contribution to variance of the interactions between groups of inputs. The Sobol indices are closely related to the ANOVA variance decomposition of a function in the case of independent input distributions [129]. In fact, we have that

$$\sum_{i=1}^{d} \sum_{i_1 < \cdots < i_s} S_{i_1,\ldots,i_s} = 1. \tag{4.5}$$

Computing all the sensitivity indices can be cumbersome. Instead the total sensitivity indices defined as :

$$S_{T,i} = \frac{\mathbb{E}\left[\mathbb{V}\left[q \mid \xi_{\sim i}\right]\right]}{\mathbb{V}\left[q\right]} \tag{4.6}$$

can be computed and they represent the contributions of ξ_i to the variance of the output including the interaction with other variables. The vector $\xi_{\sim i}$ corresponds to the vector of all entries of ξ except ξ_i. It holds that $S_i \leq S_{T,i}$. Compared to the first order indices, the total indices account for the input influence including non additive effects.

The Sobol indices are popular tools in UQ problems, but they are usually expensive to compute. If a Polynomial Chaos model is constructed can be derived they from the polynomial coefficients [102]. They can also be evaluated with dedicated sampling methods [130, 131, 132].

The Sobol indices also suffer from several interpretation limitations. First, their interpretation as sensitivity indices is only valid for independent input distributions. For dependent input distributions, dedicated indices have been proposed in [133, 134]. Second, they are well suited for univariate output models and do not generalize straightforwardly to multivariate cases [135]. Finally, they only depend on the analysis of variance, which is not a thorough measure of uncertainty, as illustrated in [136, 137]. The Sobol indices measure the contribution of one input (or group of inputs) to the deviation of q around its mean, and they are ill-suited when other statistics of q (low probability, quantiles) are targeted.

Alternative indices have been proposed in the literature have been proposed using dissimilarity measures between q and $q|X_i$ that generalize the Sobol indices using moment independent measure that lead more complete sensitivity assessments [138, 139, 140]. In [140], the computation of the indices depends on the numerical evaluation of the conditional distribution (for instance, using kernel density estimation) which is a challenging task in high dimensions. Alternatively, [141] proposes multivariate sensitivity indices based on the CDF of q and present efficient estimators associated. In [142], the authors also show that the sensitivity analysis should depend on the quantity of interest : using Sobol indices when the QoI is actually a probability or a quantile of Y is not optimal. Instead, [142] define contrast functions to construct better suited sensitivity indices.

4.1.4 Gaussian process models

Gaussian Processes can be used as surrogate models. A Gaussian Process is a collection of random variables such as any subset have a joint Gaussian distribution [120]. As a consequence, a

GP is completely defined by its mean and covariance functions. For a stochastic process denoted G indexed by $\boldsymbol{x} \in \mathbb{R}^n$ the mean and covariance are defined as :

$$\mu(\boldsymbol{x}) = \mathbb{E}\left[\mathrm{G}(\boldsymbol{x})\right] \tag{4.7}$$

and

$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}\left[\left(\mathrm{G}(\boldsymbol{x}) - \mu(\boldsymbol{x})\right)\left(\mathrm{G}(\boldsymbol{x}') - \mu(\boldsymbol{x}')\right)\right] \tag{4.8}$$

hence, following the notation in [120], we denote the GP distribution $\mathcal{GP}(\mu(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}'))$.

Regression with GPs

In machine learning and UQ, GP models are used as probabilistic approximations of generic functions $f : \Omega \subseteq \mathbb{R}^n \mapsto \mathbb{R}^m$. Hence in this model, f is an unknown function viewed as a Gaussian process G we seek to learn. For simplicity, we restrict the presentation to the case of scalar functions f, that is, $m = 1$. Let $\mathcal{X} = \{(\boldsymbol{x}_i, f(\boldsymbol{x}_i)), i = 1 \dots p\}$ be the training set, we wish to compute the distribution $\mathrm{G}(\boldsymbol{x}^*)|\mathrm{G}(\boldsymbol{x}_1) = f(\boldsymbol{x}_1) \dots \mathrm{G}(\boldsymbol{x}_p) = f(\boldsymbol{x}_p)$. In this work we assume that the Gaussian process mean function is zero. Since the joint distribution $(\mathrm{G}(\boldsymbol{x}^*), \mathrm{G}(\boldsymbol{x}_1), \dots \mathrm{G}(\boldsymbol{x}_p))$ is Gaussian distributed with mean 0 and covariance

$$\begin{pmatrix} k(\boldsymbol{x}^*, \boldsymbol{x}^*) & \mathbf{k}_{\mathcal{X}}^T \\ \mathbf{k}_{\mathcal{X}} & K_{\mathcal{X}} \end{pmatrix} \tag{4.9}$$

where $\mathbf{k}_{\mathcal{X}}(\boldsymbol{x}) = (k(\boldsymbol{x}^*, \boldsymbol{x}_1) \cdots k(\boldsymbol{x}^*, \boldsymbol{x}_p))^T$ and $K_{\mathcal{X}i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, then the conditional distribution $\mathrm{G}(\boldsymbol{x}^*)|f(\boldsymbol{x}_1) \dots f(\boldsymbol{x}_p)$ is Gaussian distributed with mean

$$\mu_{\mathcal{X}}(\boldsymbol{x}^*) = \mathbf{k}_{\mathcal{X}}(\boldsymbol{x}^*) K_{\mathcal{X}}^{-1} \mathbf{f}_{\mathcal{X}}, \tag{4.10}$$

with $\mathbf{f}_{\mathcal{X}} = [f(\boldsymbol{x}_1) \dots f(\boldsymbol{x}_p)]$ and variance

$$\sigma^2_{\mathcal{X}}(\boldsymbol{x}^*) = k(\boldsymbol{x}^*, \boldsymbol{x}^*) - \mathbf{k}_{\mathcal{X}}(\boldsymbol{x}^*) K_{\mathcal{X}}^{-1} \mathbf{k}_{\mathcal{X}}(\boldsymbol{x}^*)^T. \tag{4.11}$$

The posterior mean in Eq.(4.10) is the best prediction (in the mean squared sense) of $f(\boldsymbol{x}^*)$ based on the observations we have. In fact, for the noise free construction above, we have $\mu_{\mathcal{X}}(\boldsymbol{x}_p) = y_p$ for all $\boldsymbol{x}_p \in \mathcal{X}$, hence the GP interpolates the training points. In some cases, the observations may be noisy and instead of observing directly $f(\boldsymbol{x}_i)$ we only have access to $y_i = f(\boldsymbol{x}_i) + \epsilon$ with ϵ being a zero mean Gaussian random variable with variance σ. In this case, the distribution $(\mathrm{G}(\boldsymbol{x}^*)|\mathrm{G}(\boldsymbol{x}_1) = y_1 - \epsilon, \dots \mathrm{G}(\boldsymbol{x}_p) = y_1 - \epsilon)$ is given by :

$$\mu_{\mathcal{X}}(\boldsymbol{x}^*) = \mathbf{k}_{\mathcal{X}}(\boldsymbol{x}^*)(K + \sigma^2 I)_{\mathcal{X}}^{-1} \mathbf{y}_{\mathcal{X}}, \tag{4.12}$$

$$\sigma^2_{\mathcal{X}}(\boldsymbol{x}^*) = k(\boldsymbol{x}^*, \boldsymbol{x}^*) - \mathbf{k}_{\mathcal{X}}(\boldsymbol{x}^*)(K + \sigma^2 I)_{\mathcal{X}}^{-1} \mathbf{k}_{\mathcal{X}}(\boldsymbol{x}^*)^T, \tag{4.13}$$

with $\mathbf{y}_{\mathcal{X}} = [y_1 \dots y_p]$. Contrary to the noise free case, the noisy case equation (4.12) show that the mean GP does not interpolate the observations.
In the previous equations, the mean function, also called prior mean is set to 0. In some studies [143, 123], the mean function is defined as a regression over a set of basis functions:

$$\mu(\boldsymbol{x}) = h(\boldsymbol{x})^T \beta \tag{4.14}$$

where $h(\boldsymbol{x})$ is a vector of basis functions (for instance polynomials) and β a vector of coefficients that are inferred from the data. Details on how to incorporate basis functions to the GP prior mean are given in [120]. In this work only zero prior mean functions are considered.

Classification with GPs

In the previous paragraph, GP based surrogate models are constructed for continuous functions. In some case, however, the output of the model M to approximate can be discrete. For instance, in the simulator constructed in Chapter 3, the survivability solver returns a discrete output : the fragment burnt up in the atmosphere or it survived. Using a continuous regression model is clearly inappropriate to learn a surrogate model of the survivability solver. Instead, a classifier is better suited. The machine learning community developed a vast number of techniques for classification and the reader is referred to the reference books [116, 144] for a detailed review of such techniques. In this paragraph we detail the use of Gaussian processes for binary classification. One advantage of GPs in classification is the probabilistic interpretation of the prediction and robustness of the predictions. In this section, we wish to learn a function $C : \Omega \subset \mathbb{R}^n \mapsto \{0, 1\}$.

The GP classifier defines a membership probability π on Ω as the predicted probability that C returns 1. The GP classifier membership probability is the composition of a sigmoid function σ and a latent GP denoted l. The sigmoid function is a strictly increasing function that takes the output of l as input and returns a value between 0 and 1. Classical sigmoid functions are the probit function that is the cumulative distribution function (CDF) of a standard Gaussian and the logistic function defined as :

$$\sigma_{\log}(z) = \frac{1}{1 + \exp(-z)} \tag{4.15}$$

In this work, the probit function is preferred as we will see that it yields analytical expressions for the predicted probability.

The idea behind this construction, is to turn a GP prediction valued in $\mathbb{R}$ into a probability between 0 and 1. Hence the sigmoid σ needs to respect the point symmetry $\sigma(x) = 1 - \sigma(-x)$. This approach carries ties with the Support Vector Machine (SVM) classifier coupled with the probabilistic interpretation presented in [145]. Hence, a GP classifier defines a membership probability π as :

$$\pi(x) = \sigma(l(\boldsymbol{x})). \tag{4.16}$$

As l, π is a stochastic process but it is not Gaussian distributed due to the non linearity of σ. The average membership probability is defined as :

$$\bar{\pi}(x) = \int_l \sigma(l(\boldsymbol{x}))p(l(\boldsymbol{x}))dl. \tag{4.17}$$

As for the regression model, we consider the set of observations $\mathcal{X} = (\boldsymbol{x}_i)$. Conditioned on $\pi(\boldsymbol{x}_i)$, an observation $C(\boldsymbol{x}_i)$ is a realization of a Bernoulli trial with parameter $\pi(\boldsymbol{x}_i)$.

We wish to compute the posterior distribution $\bar{\pi}_{\mathcal{X}}(\boldsymbol{x}^*)$ defined as :

$$\bar{\pi}_{\mathcal{X}}(\boldsymbol{x}^*) = \int_{l^*} \sigma(l^*)p(l^*|\mathcal{X})dl^*. \tag{4.18}$$

Where $l^*|\mathcal{X} = l(\boldsymbol{x}^*)|\mathcal{X}$ is a latent GP prediction at test location l^*. It is evaluated using :

$$p(l^*|\mathcal{X}) = \int p(l^*|\mathcal{X}, \mathbf{l})p(\mathbf{l}|\mathcal{X})d\mathbf{l}, \tag{4.19}$$

with $\mathbf{l} = (l(\boldsymbol{x}_1), \dots l(\boldsymbol{x}_m))$. In the regression case, the predictive distribution $l^*|\mathcal{X}$ was obtained analytically by conditioning over a series of Gaussian observations. In the classification case, it is not the case since the observations are not Gaussian distributed. To see that, we apply Bayes rule to get :

$$p(\mathbf{l}|\mathcal{X}) = \frac{\prod_i^m p(C(\boldsymbol{x}_i)|l(\boldsymbol{x}_i))p(\mathbf{l})}{p(\mathcal{X})}, \tag{4.20}$$

assuming independent observations. While $p(l(\boldsymbol{x}_i))$ is Gaussian distributed, $p(C(\boldsymbol{x}_i)|l(\boldsymbol{x}_i))$ is a Bernoulli. As a result, the estimation of the posterior of the Gaussian process $p(l|\mathcal{X})$ is not analytically tractable. One approach to numerically evaluate (4.20) is to use Monte Carlo Markov Chain MCMC. Alternatively, approximate methods such as the Laplace approximation [146] or expectation propagation algorithms [147] are less expensive. Both the Laplace approximation and expectation propagation rely on an approximation of the posterior defined in Eq. (4.20). The Laplace approximation uses a Gaussian approximation of $p(l|\mathcal{X})$ with mean the quantity $\arg\max_l p(l|\mathcal{X})$ and covariance, the Hessian of the negative log posterior (see [120] for implementation details). The expectation propagation approach is a general tool for deriving posterior distributions that can be applied to approximate $p(l|\mathcal{X})$ [120].

In this work, the expectation propagation method implemented in [148] is used. Once the posterior of the GP l is computed, the other difficulty is to estimate the integral in eq. (4.19). If the probit function is chosen for σ the integral in eq. (4.19) can be solved analytically [120].

Once the posterior distribution is computed, the prediction (0 or 1) is based on the maximum a posteriori. In particular, if $\bar{\pi}_{\mathcal{X}}(\boldsymbol{x}^*) > 0.5$ then we predict 1 and 0 otherwise.

Covariance functions

The covariance function plays a significant role in the prediction as it contains the prior knowledge about the relationship between input points. A basic intuition is that input points that are close from each other should yield similar results and this information should be encoded in the covariance function. This notion is closely related to the function smoothness for instance. In other words, the covariance function characterizes the amount of information that can be extracted from previous observations to make predictions at unseen locations. The choice of covariance function is hence crucial to encode the right *a priori* information concerning the function we seek to learn in order to make the learning process more efficient. In this paragraph, we review the main properties of kernels and standard kernels used in the literature.

A covariance function or kernel is a mapping from $\Omega \times \Omega$ to $\mathbb{R}$, $(\boldsymbol{x}, \boldsymbol{x}') \mapsto k(\boldsymbol{x}, \boldsymbol{x}')$ that is symmetric ($k(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{x}', \boldsymbol{x})$). In order to be a valid kernel, $k(.,.)$ must be positive semi-definite [149]. Most kernels used for constructing surrogate models are stationary, therefore, they only depend on the difference $\boldsymbol{x} - \boldsymbol{x}'$. When the kernel only depends on the distance between $\boldsymbol{x}$ and $\boldsymbol{x}'$ it is said to be isotropic.

As mentioned previously, the smoothness of the GP prediction is encoded in the covariance function. For instance, a GP G with covariance function $k(.,.)$ is continuous (in the mean square sense) at location $\boldsymbol{x}$ if and only if $k(.,.)$ is continuous at $(\boldsymbol{x}, \boldsymbol{x})$. More generally, the derivability of f is given by the derivability of the kernel. This property can guide in the choice of the right kernel based on prior knowledge about the function smoothness.

The square exponential (SE) kernel for instance defined as

$$k_{SE}(\boldsymbol{x}, \boldsymbol{x}') = \sigma_c \exp\left(-\sum_i^d \frac{(x_i' - x_i)^2}{l_i^2}\right) \tag{4.21}$$

is infinitely smooth and so is the GP associated. For time dependent processes, this property is not necessarily a good thing [150] but for surrogate modelling, it is a classical choice. The parameters σ_c and $\boldsymbol{l} = (l_1 \dots l_n)$ are usually a priori unknown positive parameters, estimated using the training data.

Alternatively, the exponential (or Ornstein Uhlenbeck) kernel:

$$k_E(\boldsymbol{x}, \boldsymbol{x}') = \sigma_c \exp\left(-\sum_i^d \frac{|x_i' - x_i|}{l_i}\right) \tag{4.22}$$

is continuous but not derivable.

For intermediately smooth function, Stein reccommands using the Materń kernel [150] defined as

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sigma_c \frac{2^{1-\nu}}{\Gamma(\nu)} \prod_i^d \left(\sqrt{2\nu}\frac{|x_i' - x_i|}{l_i}\right)^{\nu} K_\nu \left(\sqrt{2\nu}\frac{|x_i' - x_i|}{l_i}\right) \tag{4.23}$$

with ν,l_1, ..., l_n and σ_c positive parameters, and K_ν is the modified Bessel function [120]. The regularity of Materń kernels is related to the value of ν. The GP f is k times differentiable if and only if $\nu > k$. Usually half integer values of ν are selected since the kernel becomes the product of a polynomial and an exponential. Note that the exponential kernel and the SE kernel are special cases of the Matern kernel for $\nu = 1/2$ and $\nu \longrightarrow \infty$ The smoothness of the GP has a strong influence on the samples. As an illustratin, sample paths from exponential GP (continuous), Matern kernel GPs with $\nu = 3/2$ (once derivable) and SE kernel GP are represented on fig. 4.2,

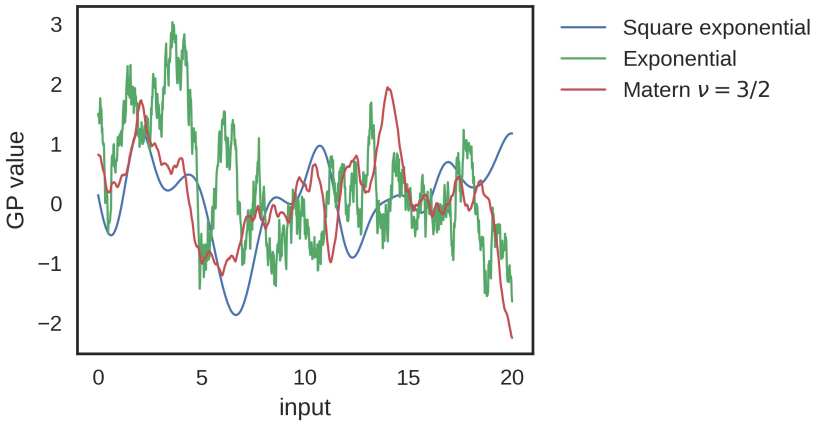


Figure 4.2: Sample paths from GP constructed with three different kernels

All the kernels presented so far are defined with the hyper-parameters σ_c called the process variance and the length scales $\boldsymbol{l} = (l_1 \dots l_n)$. The values of the process variance and length scales significantly change the behaviour of the GP. The length scales characterize the variability of the GP with respect to the inputs. A short length scale in one input dimension implies that GP is expected to have high variability in that direction. Furthermore, the amount of information available from the neighbouring training points decreases with the length scale. When all the length scales are equal, the kernel is isotropic. The parameter σ_c characterizes the amplitude of the variations of a GP. In the case of noisy observations, a third parameter σ called observation noise, quantifies our degree of confidence in the observations. While the regularity of the function can be decided a priori based on prior knowledge of the function to approximate, the values of σ_c, σ and $\boldsymbol{l}$ are usually fitted to the data. In this work, the hyper parameters σ_c, σ and $\boldsymbol{l}$ are fitted using the marginal likelihood defined as :

$$\log p(\boldsymbol{y}|\mathcal{X}) = \frac{1}{2}\boldsymbol{y}(K + \sigma^2 I)^{-1}\boldsymbol{y} - \frac{1}{2}\log(\det(K + \sigma^2 I)) + \frac{n}{2}\log(2\pi). \tag{4.24}$$

where $\boldsymbol{y} = (f(x_1) + \epsilon_1, \ldots, f(\boldsymbol{x}_p) + \epsilon_p)$. The dependency of $\log p(\boldsymbol{y}|\mathcal{X})$ in $\boldsymbol{l}$ and σ_c comes from K. In this work we choose the σ_c^*, σ^* and $\boldsymbol{l}^*$ that maximize $\log p(\boldsymbol{y}|\mathcal{X})$. The optimization is performed using modified Newton methods and multiple restarts as the likelihood derivatives can be computed analytically (see [120] for details).

Stationary kernels are standard choices in UQ, but non-stationary kernels can also be used. Examples of non-stationary kernels are the polynomial kernels, Gibbs' kernel or periodic kernels (see [120] for details). Moreover, products and sums of valid covariance functions are valid covariance functions. For many industrial applications, the predictive ability of the surrogate model can be improved with the right choice of kernel. For additive functions of the form:

$$f(\boldsymbol{x}) = c + \sum_{i=1}^{d} f_i(x_i) \tag{4.25}$$

The selected kernel should also be additive such that the stochastic process associated is a linear combination of independent one-dimensional Gaussian Processes. As suggested in [151], in this work, we often use a sum of one-dimensional kernels for the additive part of the model plus a full kernel to permit interactions between variables.

Eigenfunction decomposition of GPs

A Bayesian generalized linear regression model with Gaussian weights is a Gaussian Process by construction but it turns out that GP regression can be interpreted as a Bayesian linear regression and a possible regression basis are the kernel eigenfunctions [120]. A function $\phi : \Omega \mapsto \mathbb{R}$ that obeys the Fredholm integral equation :

$$\lambda\phi(\boldsymbol{x}) = \int_{\Omega} k(\boldsymbol{x}', \boldsymbol{x})\phi(\boldsymbol{x}')d\nu(\boldsymbol{x}) \tag{4.26}$$

is an eigenfunction of $k(.,.)$ associated with the eigenvalue λ. In general there are infinitely many eigenfunctions, otherwise the kernel is said to be degenerate. The eigenfunctions and eigenvalues are labelled such that $\lambda_1 \geq \lambda_2 \ldots$. The Karhunen-Loéve expansion KLE is a series expansion of a stochastic process into the eigenfunctions and eigenvalues of the kernel. The expansion is exact such that for a GP $\mathrm{G}(\boldsymbol{x})$ with kernel function $k(\boldsymbol{x}', \boldsymbol{x})$ we have :

$$\mathrm{G}(\boldsymbol{x}) = \mu(\boldsymbol{x}) + \sum_{i=1}^{\infty} \sqrt{\lambda_i}\phi_i(\boldsymbol{x})\zeta_i \tag{4.27}$$

where $(\zeta_i)_{i>0}$ is a set of independent Gaussian variables with zero mean and unitary variance and $\mu(\boldsymbol{x})$ the GP mean function. Note that the KLE is also defined for arbitrary second-order stochastic processes but the distribution of the $(\zeta_i)_{i>0}$ is not necessarily Gaussian and independent. The KLE can be truncated to approximate G with a finite sum of m terms:

$$\mathrm{G}(\boldsymbol{x}) \simeq \mu(\boldsymbol{x}) + \sum_{i=1}^{m} \sqrt{\lambda_i}\phi_i(\boldsymbol{x})\zeta_i \tag{4.28}$$

The truncated KLE is optimal with respect to the integrated mean square error and hence is often used to represent a GP with only a limited number of parameters. Several methods for computing the KLE of a stochastic process exist and are discussed in the review paper [152] or the reference book [102]. A classical method used in this work consists in estimating the

eigenfunctions and eigenvalues by approximating the integral in Eq. (4.26) with a quadrature rule or Monte Carlo estimation [152]:

$$\lambda_i \phi_i(\boldsymbol{x}) \simeq \frac{1}{M} \sum_{j=1}^{M} \phi_i(\boldsymbol{x}_j) k(\boldsymbol{x}_j, \boldsymbol{x}) \tag{4.29}$$

where $(\boldsymbol{x}_j)_{1 \leq j \leq M}$ are Monte Carlo samples. Evaluating Eq. (4.29) at $\boldsymbol{x} = \boldsymbol{x}_j$, Eq. (4.29) becomes an eigenvalue problem

$$K \boldsymbol{u}_i = \tilde{\lambda}_i \boldsymbol{u}_i \tag{4.30}$$

where $K_{l,k} = k(\boldsymbol{x}_l, \boldsymbol{x}_k)$ and $\boldsymbol{u}_i = \sqrt{M}(\phi_i(\boldsymbol{x}_1), \ldots, \phi_i(\boldsymbol{x}_M))^T$ and $\tilde{\lambda}_i = M\lambda_i$. Once Eq. (4.30) is solved for $\boldsymbol{u}_i$ and $\tilde{\lambda}_i$, λ_i and the $(\phi_i(\boldsymbol{x}_j))_{1 \leq j \leq M}$ can be derived. The ϕ_i can then be interpolated between the $(\phi_i(\boldsymbol{x}_j))_{1 \leq j \leq M}$ using Nystrom method for instance [152].

Relationship with Kriging

Gaussian processes have been used in the geostatistics field for over half a century (see [153]) in a non parametric regression method called Kriging (named after the pionner work of D. Krige [154]). The Kriging main idea is to approximate the mapping f with a GP and a regression term :

$$F(\boldsymbol{x}) = \sum_{i}^{m} \beta_i h_j(\boldsymbol{x}) + Z(\boldsymbol{x}) \tag{4.31}$$

where (h_j) are regression functions, (β_j) are the weights associated and $Z(\boldsymbol{x})$ a zero mean GP. The objective is to build the best unbiased linear predictor of f based on the training set $\mathcal{X}$. The predictor is constructed by imposing the unbiasedness condition while minimizing the mean square prediction error (see [119, 150] for detailed derivation). The Kriging predictor yields the same results as the GP predictor we presented earlier, but the construction is different. Kriging approaches have then been adapted to UP problems using sparse PCE [112, 123] or nested polynomial [155] as regression functions.

4.1.5 Design of experiments

The performance of the GP surrogate model is closely related to the quality of the training set $\mathcal{X}$. In this section, we develop two classes of designs of experiments. The first class concerns model-free designs where no a priori information concerning the function to approximate is known. The second class includes active learning techniques also called model-based designs.

Model-free design of experiments

The space filling designs are model-free approaches that aim at optimizing the coverage of the input space $\Omega \subset \mathbb{R}^n$ with a finite set of points of Ω denoted $\mathcal{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$. A complete review of space filling designs can be found in [156, 13], in this paragraph we introduce a limited number of strategies. Most space filling designs are constructed in bounded spaces such as $[0, 1]^n$, and in this paragraph, unless stated otherwise $\Omega = [0, 1]^n$.

We define $d_{i,j}(\mathcal{X}) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|$ where $\boldsymbol{x}_j, \boldsymbol{x}_i$ are elements of $\mathcal{X}$. The *maximin design criterion* is a DoE strategy that attempts to spread out the elements of $\mathcal{X}$ as much as possible. We define

$$Q_{Mm}(\mathcal{X}) = \min_{i \neq j} d_{i,j}(\mathcal{X}). \tag{4.32}$$

A maximin optimal training set $\mathcal{X}^*$ maximizes the quantity $Q_{Mm}(\mathcal{X}^*)$. See on fig. 4.3 for an illustration in the unit square. While the maximin design aims at maximizing the distance between training points, the *minimax design criterion* aims at minimizing the distance between any $\boldsymbol{x}$ in Ω and its closest point in $\mathcal{X}$. We define :

$$Q_{mM}(\mathcal{X}) = \max_{\boldsymbol{x}\in\Omega} \min_{\boldsymbol{x}_i\in\mathcal{X}} \|\boldsymbol{x} - \boldsymbol{x}_i\| \tag{4.33}$$

A minimax design $\mathcal{X}^*$ minimizes the quantity $Q_{mM}(\mathcal{X}^*)$. An example of minimax design is given in fig. 4.3. The maximization of Q_{Mm} and the minimization of Q_{mM} are not straightforward. Algorithms and numerical methods for implementing minimax and maximin methods are presented in [156]. The maximin and minimax method main drawback is that they offer no guaranty that projections on subspaces of Ω conserve the same property of large distance between training points and optimal space coverage. To remedy this problem, Latin Hypercube Sampling LHS can be used. Initially introduced by [157], LHS was initially designed for optimized sampling but is now widely used in the design of experiments as its projection in one-dimensional subspaces yields the optimal maximin design. For a given sample size, there a finite number of possible LHS designs. This property led to interesting works trying to select the optimal design for a given discrepancy criterion [158] or space filling design [159]. Note only the one dimensional projections of the LHS are optimal space filling designs but carry no guaranty for higher dimensional subspaces. In [160], orthogonal arrays based LHS are introduced to improve the properties of space filling designs in higher subspaces.

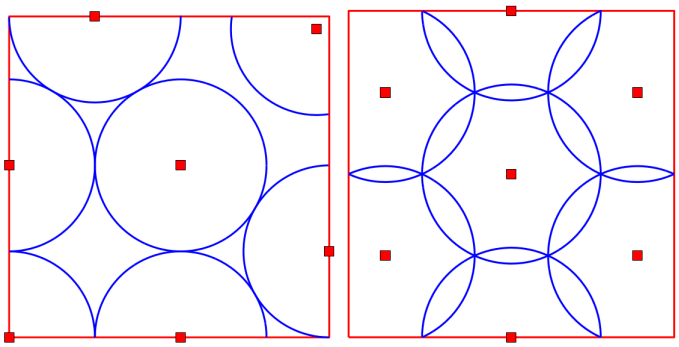


Figure 4.3: Maximin design (left) and Minimax design (right) taken from [13]

The designs proposed earlier are well suited for hypercubes of $\mathbb{R}^d$ with finite support distributions but their extension to infinite spaces such as $\mathbb{R}^d$ is not straightforward, especially when the input distributions are not independent. In more complex cases where the distribution of ξ is not defined analytically (for instance in Data-driven uncertainty propagation), clustering based approaches are effective approaches. They propose a space-filling criterion based on the minimization of the distortion :

$$D(\mathcal{X}) = \int_{\Omega} \min_{\boldsymbol{x}_i\in\mathcal{X}} \|\boldsymbol{x} - \boldsymbol{x}_i\| \, p_X(\boldsymbol{x}) d\boldsymbol{x}. \tag{4.34}$$

In practice, the integral is evaluated with MC samples. This criterion is proposed by [161, 162, 163] under the name of principal points and Fast Flexible space Filling design (FFF). It carries analogy with vector quantization [164] and k-means clustering [116] that also aim at minimizing the same distortion. Note the difference with the minimax criterion in eq. (4.33) that minimizes the maximum of $\min_{\boldsymbol{x}_i\in\mathcal{X}} \|\boldsymbol{x} - \boldsymbol{x}_i\|$ while FFF minimizes the average of $\min_{\boldsymbol{x}_i\in\mathcal{X}} \|\boldsymbol{x} - \boldsymbol{x}_i\|$. In a

sense, the minimax criterion is a worst case scenario criterion while FFF is an averaged criterion. The main advantages of clustering based designs are that they can be used with dependent data generated distributions with convex support. Finding the minimum of $D(\mathcal{X})$ is NP hard but efficient heuristic approaches can be used to find a local minimum of the distortion. The most classical ones are Lloyd's algorithm [165] and Kohonen's algorithm [166].

Active learning for regression problems

The design of experiment approaches presented in the previous paragraph are model free. They make no assumptions regarding the structure of f. In this section, we present some classical model-based design of experiment methods. Such approaches are referred to in the literature as Active Learning AL or adaptive sampling. A natural way to include model information in the construction of DoE is to rely on the GP framework presented earlier. Function characteristics (smoothness, length scale, additivity) can be encoded in the kernel or learned from previous observations and used to construct training sample sets. We use the notation from the previous paragraphs where G is a GP and $\mathcal{X}$ is the training set associated. The posterior distribution is given by $\mu_{\mathcal{X}}(\boldsymbol{x})$ from Eq. (4.10) and $\sigma_{\mathcal{X}}(\boldsymbol{x})$ from Eq. (4.11) at test location $\boldsymbol{x} \in \Omega$. Aside from uncertainty propagation, active learning strategies have been developed in surrogate model based optimization [167], reliability problems [168] although in this paragraph, we focus on AL strategies for UP. An interesting review on AL methods can be found in [169].

The large set of AL methods for UP aims at minimizing L_2 error defined as :

$$E = \left(\int_{\Omega} \left(f(\boldsymbol{x}) - \mu_{\mathcal{X}}(\boldsymbol{x}) \right)^2 d\boldsymbol{x} \right)^{1/2} \tag{4.35}$$

where f is the model and $\mu_{\mathcal{X}}(\boldsymbol{x})$ the mean posterior of G. As f is unknown, the L_2 error cannot be evaluated directly. Instead, the predictive variance $\sigma^2(\boldsymbol{x})$ of the GP is used as a surrogate of $(f(\boldsymbol{x}) - \mu_{\mathcal{X}}(\boldsymbol{x}))^2$. Using the predictive variance as an estimate of the L_2 error is the backbone of most active learning strategies. In [170, 120], this approach is justified when computing the average L_2 error defined as

$$E_g(\boldsymbol{x}) = \mathbb{E}\left[\left(f(\boldsymbol{x}) - \mu_{\mathcal{X}}(\boldsymbol{x}) \right) \right)^2 \Big] . \tag{4.36}$$

The average is computed over the unknown model f we seek to learn. In [170, 120], the distribution of f is given by the GP prior distribution such that [120]

$$E_g(\boldsymbol{x}) = \sigma^2_{\mathcal{X}}(\boldsymbol{x}). \tag{4.37}$$

In the rest of this presentation, we take for granted that the predictive variance is a reasonable estimate of the L_2 error. We introduce four classical AL criteria. The Maximum Mean Square Predictive Error MMSPE, the Integrated Mean Square Predictive Error IMSPE based on reducing the predictive mean square error and two other criteria based on information theory quantities, namely the entropy and the mutual information.

The MMSPE criterion is a minimax approach defined as [171] :

$$Q_{MMSPE}(\mathcal{X}) = \max_{x \in \Omega} \sigma_{\mathcal{X}}(\boldsymbol{x}) \tag{4.38}$$

where the optimal training set $\mathcal{X}^*$ is such that :

$$\mathcal{X}^* = \arg\min_{\mathcal{X}} Q_{MMSPE}(\mathcal{X}) \tag{4.39}$$

Finding the optimal training sample set solving Eq. 4.39 is a formidable task. It involves a $d \times p$-dimensional optimization with numerous local minima. Considering a finite set of potential candidates can significantly reduce the computational burden. Numerical methods based on a quasi-Newton optimization, exchange algorithms, and simulated annealing have been proposed in[172, 173, 174, 175]. A second challenge raised by the resolution of Eq. (4.39) is related to the kernel properties. The optimal training set depends on the kernel structure and hyper-parameters that are usually estimated *a posteriori*, from the data. On the way to alleviate those difficulties is to use sequential heuristics to obtain a suboptimal solution of (4.39). In [176], the author proposes a sequential strategy based on the maximum entropy reduction. The sequence of training samples is generated as follow. Define $\mathcal{X}^p = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_p\}$ the next training point x_{p+1} is solution of

$$\max_{\boldsymbol{x} \in \Omega} \sigma_{\mathcal{X}^p}(\boldsymbol{x}) \tag{4.40}$$

Compared to (4.39), the optimization problem in Eq. (4.40) is only d-dimensional. Overall, the procedure requires the resolution of p optimizations in d dimensions. Moreover, for practical applications, the $\mathcal{X}$ is usually initialized with a model-free criterion (for instance LHS) and is subsequently enriched with (4.40). One advantage of the method is that the kernel hyper-parameters can be learned at each training point addition.

The MMSPE criterion in the Gaussian case is also called D-optimal and is equivalent to entropy reduction approaches. Let consider an arbitrary discretization of Ω denoted $\mathcal{V}$. The differential entropy at unobserved locations ($\mathcal{V} \setminus \mathcal{X}$) conditioned on $\mathcal{X}$ is defined as [177] :

$$H\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X}) | \mathrm{G}(\mathcal{X})\right) = \int p\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X}) | \mathrm{G}(\mathcal{X})\right) \log p\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X}) | \mathrm{G}(\mathcal{X})\right) d\mathrm{G}(\mathcal{V} \setminus \mathcal{X}) d\mathrm{G}(\mathcal{X}) \tag{4.41}$$

where $\mathrm{G}(\mathcal{V} \setminus \mathcal{X})$ is the GP G evaluated at $\mathcal{V} \setminus \mathcal{X}$ and $\mathrm{G}(\mathcal{X})$ is the GP G evaluated at $\mathcal{X}$. Minimizing this quantity implies minimizing the uncertainty at unseen locations $\mathcal{V} \setminus \mathcal{X}$ given the seen locations. Since $H\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X}) | \mathrm{G}(\mathcal{X})\right) = H\left(\mathrm{G}(\mathcal{V})\right) - H\left(\mathrm{G}(\mathcal{X})\right)$, we see that minimizing $H\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X}) | \mathrm{G}(\mathcal{X})\right)$ is equivalent to maximizing $H(\mathrm{G}(\mathcal{X}))$. As for the MMSPE criterion, finding $\arg\max_{\mathcal{X}} H(\mathrm{G}(\mathcal{X}))$ is extremely challenging and sequential heuristics are usually preferred. For a given $\mathcal{X}^p = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_p\}$ we define :

$$\boldsymbol{x}_{p+1} = \arg\max_{x \in \mathcal{V}} H\left(\mathrm{G}(\boldsymbol{x}) | \mathrm{G}(\mathcal{X}^p)\right) \tag{4.42}$$

and since G is a GP, $H\left(\mathrm{G}(\boldsymbol{x}) | \mathrm{G}(\mathcal{X}^p)\right) = \frac{1}{2} \log\left(2\pi e \sigma^p_{\mathcal{X}}(\boldsymbol{x})\right)$. Hence the optimization problems in (4.40) and (4.42) are equivalent. The MMSPE criterion is relatively inexpensive to run in its sequential form, but it tends to perform well only with low input dimension problems. Several studies assert that the MMSPE tends to add training points on the boundaries of the domain to spread out the training points as much as possible [177, 178, 179]. This property is amplified in high dimensions such that the training points are located at the corners of Ω. This property is detrimental for the GP performance as the training points near the domain boundary tend to be less informative for unobserved locations [177]. This phenomenon already mentioned in [177, 179, 180] was also observed in preliminary tests performed for this work.

In [171], an alternative criterion is proposed based on the Integrated Mean Square Predictive Error (IMSPE). Instead of minimizing the maximum predictive error, the IMSPE active learner aims at minimizing

$$Q_{IMSPE}(\mathcal{X}) = \int_{\Omega} \sigma_{\mathcal{X}}(\boldsymbol{x}) d\boldsymbol{x} \tag{4.43}$$

such that the optimal training set for the IMSPE criterion is

$$\mathcal{X}^* = \arg\min_{\mathcal{X}} Q_{IMSPE}(\mathcal{X}) \tag{4.44}$$

As for the MMSPE criterion, solving (4.44) is a challenging task and classical approaches consist in finding the optimal $\mathcal{X}$ in a finite set of candidates. Several methods based on quadrature integration of the truncated kernel [181] or quasi-Newton methods [178] have been proposed. As for the MMSPE, a classical approach is to consider a sequential resolution of (4.44). The sequential version of IMSPE is introduced in [182]. The general idea is to sequentially enrich the training set with a sample that induces the largest integrated variance reduction. At iteration p, the training set $\mathcal{X}^p$ is to be enriched with the sample x_{p+1} solution of the optimization problem

$$\int_{\Omega} \sigma_{\mathcal{X}^p}(\boldsymbol{x}) - \sigma_{\mathcal{X}^p \cup \{\boldsymbol{x}_{p+1}\}}(\boldsymbol{x}) d\boldsymbol{x} \tag{4.45}$$

By construction $\sigma_{\mathcal{X}^p}(\boldsymbol{x}) - \sigma_{\mathcal{X}^p \cup \{\boldsymbol{x}_{p+1}\}}$ is always positive and the value of $f(\boldsymbol{x}_{p+1})$ is not necessary to compute $\sigma_{\mathcal{X}^p \cup \{\boldsymbol{x}_{p+1}\}}$. Hence, the model does not have to be evaluated to solve this optimization problem. In practice, the integral is approximated by MC sampling. A priori, evaluating Eq. (4.45) scales as $(O)(p^3 n_{MC})$ where n_{MC} is the number of MC sample to evaluate the integral. This computational cost can be significantly reduced to $(O)(p^3 + p^2 n_{MC})$ (see [183]). Numerical experiment show that IMSPE based training sets do not select training points at the boundaries of the domain and hence perform better [177, 180] than MMSPE. This improvement comes with an increased computational cost when solving (4.45) instead of (4.40). An alternative method to IMSPE based on Mutual Information (MI) have also been proposed in [177, 180]. This criterion is based on the mutual information between predictions at unobserved locations and the observation :

$$I\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X}), \mathrm{G}(\mathcal{X})\right) = H\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X})\right) - H\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X}) | \mathrm{G}(\mathcal{X})\right) \tag{4.46}$$

Intuitively, the mutual information I quantifies the entropy reduction at the unobserved location (hence at the location where one wants to make predictions) induced by the training set $\mathcal{X}$. Contrary to the entropy criterion that maximizes the entropy of the training set regardless of the unobserved location predictions, the MI criterion seeks to maximize the amount of information brought by $\mathcal{X}$ to unobserved locations. The optimal $\mathcal{X}^*$ for the MI criterion maximizes the quantity $I\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X}), \mathrm{G}(\mathcal{X})\right)$. As always, maximizing $I\left(\mathrm{G}(\mathcal{V} \setminus \mathcal{X}), \mathrm{G}(\mathcal{X})\right)$ with respect to $\mathcal{X}$ is a formidable task. Sequential suboptimal algorithms are proposed in [177] and improved in [180]. They yield equivalent or slightly better results than IMSPE and MMSPE in the experiments performed by Krause *et al.* and Beck *et al.* [177, 180] but with a lower computational load.

4.1.5.1 Active learning for classification problems

Active learning classification problems have been extensively studied by the machine learning community for text classification, user preference etc. A large panel of those active learning strategies are presented in details in [144]. In this section, we only review a small subset of classification methods that are of interest for our uncertainty propagation problem. Two main classes of active learners can be identified : the likelihood criteria and the uncertainty criteria. The uncertainty criteria interpret the classifier prediction at location $\boldsymbol{x} \in \Omega$ as a Bernoulli random variable $\tilde{C}(\boldsymbol{x})$ with parameter $\bar{\pi}_{\mathcal{X}}(\boldsymbol{x})$. Uncertainty based criteria aim at minimizing the conditional entropy at the unobserved locations is

$$H\left(\tilde{C}(\mathcal{V} \setminus \mathcal{X}) | \tilde{C}(\mathcal{X})\right) = H(\tilde{C}(\mathcal{V}) - H(\tilde{C}(\mathcal{X})). \tag{4.47}$$

As in the regression case, we see that minimizing $H(\tilde{C}(\mathcal{V} \setminus \mathcal{X}) | \tilde{C}(\mathcal{X})$ corresponds to maximizing $H(\tilde{C}(\mathcal{X}))$. In [184, 185] propose to reuse a sequential heuristic that consists in adding training points where the prediction entropy is maximal. This approach however is not robust as the

prediction uncertainty is always maximal at the boundary between the two classes. To overcome this issue [184] uses a finite pool of candidates from which the training samples are selected to maximize $\mathcal{X}$. Alternatively, [186] propose to add training points that provide the largest expected reduction of an approximation of the validation error. Uncertainty based methods are natural for discriminative classifier (as the GP classifier) and perform well in refining boundaries but have limited exploratory abilities to detect new classes or regions [187]. They are also known to perform poorly when the classes are non separable.

The other class of active learners for classification uses the classifier likelihood [188]. They are particularly efficient for novelty detection (i.e. detecting new classes). More recent work suggest using multiple criteria to detect and refine classes [187]. In this work a discriminative classifier is used and the two classes are separable so that an uncertainty based criterion is preferred.

In this section, we introduced the standard methodology for uncertainty propagation, and we discussed how uncertainties can be modeled and propagated efficiently. We focused on Gaussian Process based surrogate models and dedicated approaches for the design of experiments. In the rest of this chapter, we present an original approach for constructing surrogate models of directed systems of solvers using GP.

4.2 Systems of solvers

Many engineering problems involve a multi-physics environment requiring the resolution of multiple physical phenomena. The global solution to these problems is generally obtained by coupling different solvers, each one devoted to a specific aspect of the problem. The simulator presented in chapter 3 is a good example of industrial problems solved with multiple coupled solvers. In chapter 3 the solvers used to perform rentry simulations are developed independently by teams of experts on each aspect of the physics (ARPEGE for aerothermodynamics, BL43 for trajectory propagation, ADRYANS for thermal response ...). They are subsequently coupled to form what we call in this work a *system of solvers* (SoS). Formally, we define a SoS as a set of interdependent solvers. The SoS are constructed to create a mapping from a set of *global inputs* to the quantity of interest also called *global output*. By construction, SoS form composite structures where each solver use specific numerical methods to solve distinct aspects of the physics at different computational cost and influence on the quantity of interest. In a SoS, the solvers are connected through their local inputs and outputs : the input of a solver can either be a global input of the SoS or an output of another solver. These network of dependencies requires additional software development to build the interface and dependencies between each solver.

4.2.1 Directed systems of solvers

We say that a SoS is directed if the information can only be transferred forward in the system : with respect to a particular solver, the outputs of its downstream solvers cannot be inputs of any of its upstream solvers. On the contrary, in a strongly coupled SoS the inputs of a solver can be the outputs of a downstream solver.

Without loss of generality, we shall restrict ourselves to the case of global scalar output in the following. In this work, we restrict ourselves to the particular case of *directed systems of solvers* where the solvers can be ordered along the upstream to downstream direction. Specifically, an output of a solver can only be an input of a downstream solver, such that the information (simulation results) flows in one direction only. Hence, all upstream solvers must have been run before running the downstream ones. This restriction rules out the case of strongly coupled

solvers, which must be considered as a whole. The simulator constructed in the previous Chapter 3 is directed SoS as the trajectory module coupled with the thermal module are viewed as a unique solver.

Fig 4.4 shows an example of a directed SoS. In the plot, the boxes labeled with letters represent the constituting solvers; the arrows are used to represent the inputs (arrows coming in) and outputs (arrows coming out) connecting the solvers. In this example, the solver E is an upstream solver for the solvers F, G, I and J, and a downstream solver for A, B, C, and D. We remark that the directed SoS could be also partitioned into non-overlapping blocks with boundaries corresponding to computational barriers reflecting the structure of the system. Such a block can be composed of a single or a set of solvers, with outputs of upstream blocks (or global inputs) as inputs, and outputs being inputs of downstream blocks. A key point is that solvers constituting a block can be run independently in parallel. The example depicted in Fig 4.4 illustrates the non-uniqueness of the partition of the SoS into blocks, which are represented by the dashed lines rectangles in the figure. Indeed, the solver H, belonging with the solver E to the third block, could have also been attached to the first or second blocks. In our framework, we construct a surrogate model for each solver, such that the non-uniqueness of the block decomposition is not a concern.

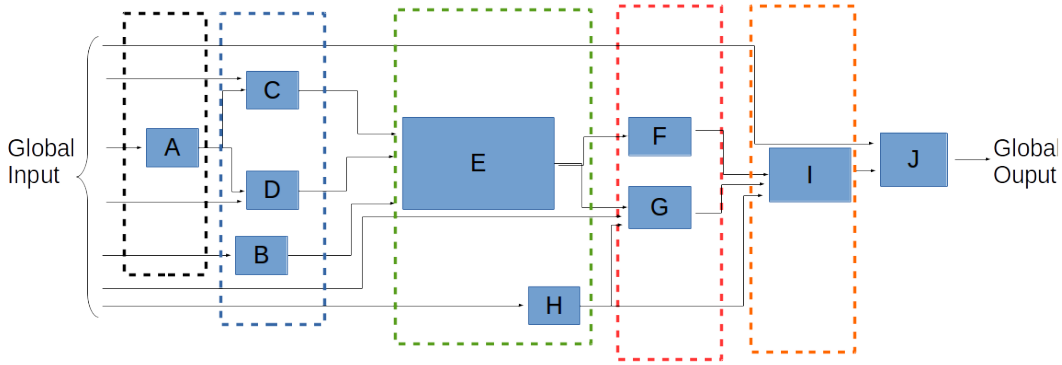


Figure 4.4: Example of directed SoS.

4.2.2 Uncertainty quantification for directed systems of solvers

The modular aspect of SoS makes them flexible and popular for industrial applications involving complex systems. Nevertheless, the application of Uncertainty Quantification methods presented in 4.1.2 to SoS is particularly challenging. First, systems of solvers commonly involve a large number of uncertain inputs. This property generally challenges the efficiency of UQ methods based on functional approximations. Second, a SoS is usually computationally expensive to run as it requires the sequential run of several solvers. These two challenges can rule out the application of standard UQ methods designed for single solvers. In this section, we propose an approach that exploits the structure of the SoS to build a surrogate model of the SoS. Two approaches are classically used in the industry to build surrogate models and quantify uncertainties in SoS: the black box approach and the fragmented approach.

In the black-box approach, the SoS is seen as a whole, and the structure of the SoS along with its internal solvers is not taken into account in the UQ analysis. A surrogate model is built in order to create a direct mapping between the uncertain global inputs and the quantities of

interest (global outputs). Within this approach, Polynomial Chaos expansions, GP models, low rank approximations are possible alternatives to construct a global surrogate as mentioned in section 4.1.2. One major drawback of these alternatives is their computational cost that can dramatically increase with the number of (global) uncertain inputs, commonly referred to as Curse of Dimensionality [103]. Further, these methods can be challenged by the highly non-linear dependencies between the global inputs and outputs induced by the structure of the SoS.

An alternative approach called the *fragmented approach* in the following, consists of building a surrogate model for each solver. In the fragmented approach, each surrogate relates the inputs of a solver to its outputs, and a prediction of the global outputs is obtained by substituting each solver with its surrogate model. This approach is common and particularly suited in situations where the individual solvers are developed, maintained and run by distinct teams. The fragmented approach has a clear advantage compared to the global black box approach when the inputs dimensionality of each solver is lower than the dimensionality of the global inputs. In this situation, it can be more effective to construct several low dimensional surrogate models, rather than constructing a single high-dimensional global one, therefore mitigating the curse of dimensionality. Besides, the individual solvers may exhibit simple mappings from inputs to outputs, where their composition may yield complex dependencies. On the other hand, the definition of the inputs probability measures to be used when constructing the individual surrogates represents a significant drawback of the *fragmented approach*. The probability measure of an input that is the output of an upstream solver is unknown a priori. One solution to this issue consists in assuming an a priori distribution for these inputs. However, proposing a distribution a priori is a difficult task: being too conservative (*e.g.* considering large input ranges) can be detrimental to the overall efficiency, when disregarding possible input values can result in large prediction errors. A possibility to overcome the difficulties in defining the input distributions is to rely on training sets resulting from a global run of entire SoS corresponding to a sample set of the global inputs. This approach ensures the consistency between the sample sets of inputs for the individual solvers. However, this approach relies on the sequential nature of the directed SoS and prevents the possibility of performing parallel runs of a solver and to focus the computational resources on particular solvers demanding larger training sets to construct their surrogate models.

Recently, systems of solvers have received interest from the UQ community trying to develop efficient UQ methods for SoS. In [189], the authors proposed a method based on importance sampling to decouple the uncertainty propagation process of individual solvers in order to gain flexibility. Other recent works focused on adapting Global Polynomial Chaos (gPC) based methods to SoS, with the challenge of deriving efficient quadrature rules on intermediate inputs with unknown distributions. Using the structure of SoS, the authors of [190] proposed a method for propagating uncertainty in a composite function by adapting the quadrature rule of intermediate inputs in the SoS, thus limiting the number of quadrature points compared to a global black box approach. This work used the recursive formula for orthogonal polynomials and Lanczos algorithms. The same authors generalized this idea in [191] to a full SoS. Their approach relies on Galerkin projection methods at intermediate layers of the SoS. By solving an optimization problem, they proposed a quadrature rule for latent variables, regularized in order to promote sparsity in the weights, thus reducing the number of quadrature points. In [192], the authors tackled the problem of strongly coupled systems. Their main idea is that the dimension of the coupling variables and the amount of information transferred from one solver to another is not as high as the actual inputs dimensionality. Consequently, they use Karhunen–Loève expansions to reduce the inputs space of each solver at each iteration, therefore

lowering the computational cost, when propagating the uncertainty through the coupled system. This idea further is used in [193], where a hybrid decomposition of the random output field is proposed. This decomposition is used to construct surrogate models of a system of solvers and perform a Bayesian optimization [194]. In [195], the authors proposed a framework for uncertainty propagation for directed SoS. Their framework applies to intrusive and non-intrusive methods such as Monte Carlo, non-intrusive polynomial chaos and Galerkin method. In [196] the framework presented in [195] is generalized to non-directed systems. Their approach relies on restriction and expansion operators adjusted to the dimension of the intermediate inputs. The UQ methods rely on a global polynomial approximation but with quadrature rules adapted to the local problems. Hence, the local quadrature rules are improved compared to the global black-box approach. From a Bayesian perspective, the authors of [197] proposed a UQ propagation framework in SoS for data assimilation in multiphysics problems.

Our approach tackles the problem from a different perspective. We introduce a new predictive model called System of Gaussian Processes (SoGP) suitable for directed systems of solvers. In our approach, a GP model is constructed for each solver of the SoS, and the global prediction is built by propagating the GPs predictions. Our framework carries similarities with Deep Gaussian Processes [198] and Multi-step ahead predictions [199], although the objectives and construction differ in our framework. A similar framework for a two-solvers problem is presented in [200]. A significant contribution of our work is the formulation of suitable criteria to design efficient adaptive training strategies. The key ideas are to simultaneously exploit the advantages of a fragmented approach (low dimensionality, flexibility) when building the surrogate model of a solver, with the use of global criteria to weight the importance of each solver on the global outputs prediction error. Specifically, a decomposition of the SoGP prediction variance is presented. It provides a ranking according to the GP model contribution to the global error that can be used in order to enhance the overall predictive capabilities of the SoGP. This decomposition leads to efficient training algorithms that identify the GP model that should be refined in order to improve the prediction of the global outputs.

4.3 System of Gaussian processes

In Section 4.2.1, we discussed the structure of the SoS considered in this work; this structure is now exploited to construct a system of Gaussian processes (SoGP) approximating the original SoS. This results will serve as a basis to propose several estimates and decompositions of the predictive variance, in Section 4.4, and derive adaptive sampling strategies in Section 4.5. The SoGP considered in this work is obtained by substituting the solvers of a directed SoS with GP models. The focus of the present section is to define the SoGP prediction, as resulting from the composition of the GP models. The composition of GPs has been studied in machine learning as the GP-based equivalent of Neural Networks (called Deep Gaussian Processes -DGP- [198]). However, our SoGP case is different from the one usually considered in machine learning:

- there are no latent variables since all intermediate variables are observed (inputs and outputs of the constitutive solvers),
- the GPs (layers) are derived from the SoS structure, and their definition is not left to the user choice (each solver has a corresponding GP).

To illustrate the prediction of SoGP output, we consider the simple system consisting of just two solvers as illustrated in Fig. 4.5. The two solvers (f_1 and f_2 respectively) are substituted with two GPs (G_1 and G_2 respectively). Even in this simple situation, the prediction using the SoGP is not unique; two possibilities can be readily proposed:

- Given the global input $x_0 \in \mathbb{R}^n$, the best prediction of G_1 (the mean $\mu_1(x_0)$) can be used as the input of G_2 to retrieve its best prediction of the QoI. That is, for the SoGP of Fig. 4.5,

$$f_2 \circ f_1(x_0) \approx \mu_2(\mu_1(x_0)) = \mu_2 \circ \mu_1(x_0).$$

  This approach is easily generalized to more complex SoGP and will be called the **composition of the averages** in the following. Note that we drop the bold letter notation to differentiate vectors and their coordinates since from no on, we will only consider vectors. We remark that this composition of averages provides a deterministic prediction of the QoI, with no characterization of its uncertainty. However, it is clear that the manipulation and propagation of deterministic values in the SoGP is computationally convenient.

- Alternatively, one can keep the whole Gaussian distribution of $X_1 = \mathrm{G}_1(x_0)$ as the input of G_2, and defines the prediction as the resulting average, namely,

$$f_2 \circ f_1(x_0) \approx \mathbb{E}\left[\mathrm{G}_2 \circ \mathrm{G}_1(x_0)\right],$$

  where $\mathbb{E}[\cdot]$ denotes the expectation operator. In the following, we call this approach the **averaged composition of GPs**. In general, the full distribution of the prediction is transmitted from a GP to the next, all along the SoGP. Note that in the case of the system shown in Fig. 4.5 it comes

$$f_2 \circ f_1(x_0) \approx \mathbb{E}\left[[\mathrm{G}_2 \circ \mathrm{G}_1(x_0)\right] = \frac{1}{\sqrt{2\pi\sigma_1^2(x_0)}} \int \mu_2(x_1) \exp\left(-\frac{(x_1 - \mu_1(x_0))^2}{2\sigma_1^2(x_0)}\right) dx_1. \quad (4.48)$$

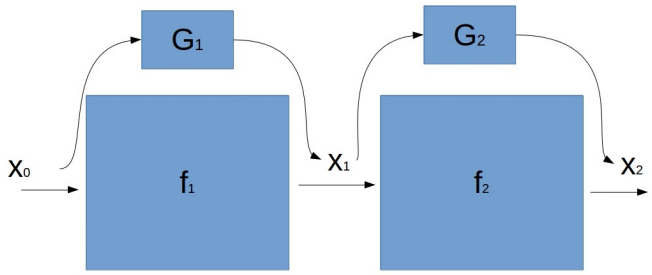


Figure 4.5: Example of 2 solvers directly chain and the corresponding SoGPs.

It is clear that, in general,

$$\mathbb{E}\left[\mathrm{G}_2 \circ \mathrm{G}_1(x_0)\right] \neq \mu_2 \circ \mu_1(x_0),$$

so that the composition of averages and the averaged composition of GPs are not equivalent. While the composition of the averages is computationally the fastest and the easiest to implement, the predictive distribution is lost because of the intermediate averaging of the GP outputs. On the contrary, the averaged composition of GPs propagates the full predictive distribution through the SoGP, therefore allowing to estimate the confidence in the predicted QoI. The main issue with this second approach is that even for just two chained solvers, the distribution of $\mathrm{G}_2 \circ \mathrm{G}_1(x_0)$ is in general not Gaussian [198, 199], preventing the derivation of explicit formulas (such as in (4.48)) for the corresponding prediction. The loss of Gaussianity when composing the GPs is due to the nonlinear character of the mapping between the inputs and outputs of a GP model. It is thus tempting to recover a Gaussian prediction using local linearizations of the GPs, around the

inputs mean value, as proposed in [199]. For instance, the example would lead to the Gaussian approximation of the composition

$$\mathrm{G}_2 \circ \mathrm{G}_1(x_0) \approx N(\mu_2(\mu_1(x_0)), \sigma_2^2 + |\mu_2'(\mu_1(x_0))|^2 \sigma_1^2(x_0)),$$

where μ_2' is the derivative of $\mu_2(x_1)$, and $N(\mu, \sigma^2)$ denotes the normal variable with mean μ and variance σ^2. We remark that the mean prediction for the linearized approach coincides with the prediction using the composition of the averages. In fact, the linearization can be seen as an approximated approach to propagate variances of the outputs along the SoGP and come-up with a Gaussian prediction of the QoI. This idea is further exploited in section 4.4.

The distribution of the averaged composition of GPs prediction can also be recast in a (high dimensional) integral of conditional probabilities. For instance, the case of the system with four solvers shown in Fig. 4.6 leads to

$$p(x_4|x_0) = \int_{x_1}\int_{x_2}\int_{x_3} p(x_4, x_3, x_2, x_1|x_0) dx_1 dx_2 dx_3 \tag{4.49}$$

$$= \int_{x_1}\int_{x_2}\int_{x_3} p(x_4, x_3, x_2|x_1) p(x_1|x_0) dx_1 dx_2 dx_3 \tag{4.50}$$

$$= \int_{x_1}\int_{x_2}\int_{x_3} p(x_4|x_3) p(x_3|x_2) p(x_2|x_1) p(x_1|x_0) dx_1 dx_2 dx_3. \tag{4.51}$$

In the previous expressions, the elementary conditional densities $p(x_i|x_{i-1})$ are all Gaussian; specifically

$$p(x_i|x_{i-1}) = \frac{1}{\sqrt{2\pi\sigma_i^2(x_{i-1})}} \exp\left[-\frac{(x_i - \mu_i(x_{i-1})^2)}{2\sigma_i^2(x_{i-1})}\right]. \tag{4.52}$$

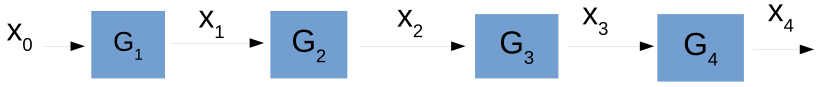


Figure 4.6: Example of SoGP (block view).

This expression of $p(x_4|x_0)$ and the Gaussian nature of the conditional densities (4.52) show that, in principle, one can accurately evaluate the averaged composition of GPs employing tensorized Gaussian quadrature rules. However, the computational cost of tensorized quadrature rules would increase exponentially with the number of chained GPs and inputs, limiting its applicability to simple trivial systems. Sparse quadrature rules could be employed to estimate at a reduced cost the high dimensional integrals, but we found more effective to proceed by Monte Carlo sampling in the present work. Indeed Monte Carlo methods are insensitive to dimensionality and can be easily applied to SoGPs with complex structures. Specifically, for the example of Fig. 4.6, one generates randomly joint samples of (X_1, X_2, X_3, X_4), using the elementary Gaussian conditional probabilities in (4.52), in order to estimate the averaged composition of GPs prediction $\mathbb{E}[X_4]$. Note that these Monte Carlo samples can also be used to estimate the variance and (non-Gaussian) density of any of the intermediate output X_i of the SoGP, and assess their respective uncertainty as further discussed in the following.

4.4 Decomposition of the predictive variance

In this section, we present the decomposition of the prediction variance of a SoGP. For simplicity, we restrict ourselves to the case of n Gaussian models G_i, directly chained one after the other, with the output of G_i being the (only) input of G_{i+1}. We shall denote $x_0 \in \Omega_0$ the global input of G_1. Further, to alleviate notational burden we shall consider that $\mathrm{G}_i : \mathbb{R} \mapsto \mathbb{R}$, although the derivations below can be easily extended to more complex situations with higher dimensional inputs and outputs (see also discussion in Section 4.4.3).

Our objective is to decompose the SoGP prediction variance into individual contributions V_i related to the model G_i and rank the importance of the GP models in view of improving the overall predictive capabilities of the SoGP. The decomposition of the SoGP prediction variance is exploited in the next section to propose dedicated training strategies. In the following, we first define the elements V_i of the decomposition (Section 4.4.1) and discuss their practical estimation (Section 4.4.2). Two approximations of the estimator of V_i, differing in their computational cost and accuracy, are subsequently proposed in Sections 4.4.2 and 4.4.2. Finally, we discuss the decomposition of the variance in the case of directed SoGPs having more general structures in Section 4.4.3.

4.4.1 Variance decomposition

For convenience, we set

$$\mathrm{G}_{j\mapsto i} := \mathrm{G}_i \circ \cdots \circ \mathrm{G}_j, \quad 1 \leq j < i \leq n, \tag{4.53}$$

$$\mathrm{G}_{j\mapsto i} := \mathrm{G}_i, \quad if j = i, \ for 1 \leq i \leq n. \tag{4.54}$$

With this notation, the predictive variance associated to $x_0 \in \Omega_0$ is $\mathbb{V}\left[\mathrm{G}_{1\mapsto n}(x_0)\right]$. To access the contributions of different solvers onto the predictive variance, we define $V_{1\mapsto i}(x_0)$ as variance of the expected prediction conditioned on $\mathrm{G}_{1\mapsto i}(x_0)$, that is

$$V_{1\mapsto i}(x_0) := \mathbb{V}\left[\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i}(x_0)\right]\right], \quad i = 1, \ldots, n. \tag{4.55}$$

The variance $V_{1\mapsto i}$ is interpreted as the variance in the (final) prediction of the global output caused by the predictive variability of $\mathrm{G}_{1\mapsto i}$, that is the SoGP up to the i-th solver. Setting $V_{1\mapsto 0}(x_0) := 0$, we note that $\{V_{1\mapsto i}\}_{i=0}^{i=n}$ forms an increasing sequence from $V_{1\mapsto 0}(x_0) := 0$ to $V_{1\mapsto n}(x_0) = \mathbb{V}\left[\mathrm{G}_{1\mapsto n}(x_0)\right]$, such that $V_{1\mapsto j}(x_0) \leq V_{1\mapsto i}(x_0)$ for $0 \leq j \leq i \leq n$ and $\forall x_0 \in \Omega_0$. Therefore, we define the predictive variance incurring to the G_i as

$$V_i(x_0) := V_{1\mapsto i}(x_0) - V_{1\mapsto i-1}(x_0) \geq 0, \quad i = 1, \ldots, n. \tag{4.56}$$

Observing that $\mathbb{E}\left[\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i}(x_0)\right]\right] = \mathbb{E}\left[\mathrm{G}_{1\mapsto n}(x_0)\right]$, Eq. (4.55) becomes

$$V_{1\mapsto i}(x_0) = \mathbb{E}\left[\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i}(x_0)\right]^2\right] - \mathbb{E}\left[\mathrm{G}_{1\mapsto n}(x_0)\right]^2,$$

and the expression of V_i can be recast to

$$\begin{aligned} V_i(x_0) &= V_{1\mapsto i}(x_0) - V_{1\mapsto i-1}(x_0) \\ &= \mathbb{E}\left[\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i}(x_0)\right]^2\right] - \mathbb{E}\left[\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i-1}(x_0)\right]^2\right], \end{aligned} \tag{4.57}$$

setting $\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto 0}(x_0)\right] := \mu_n \circ \ldots\ldots \mu_1(x_0)$.

4.4.2 Practical estimation

In this section, we propose three numerical methods with different degree of accuracy and computational cost for evaluating the terms of the variance decomposition.

Monte Carlo estimation

To estimate the partial variances V_i at a given $x_0 \in \Omega_0$, one could consider computing first the variances $V_{1\mapsto i}$ using (4.55), that is through the estimation of the variance of conditional expectations. This approach would lead to a stratified Monte Carlo (MC) method with nested loops on samples. Although the computations would involve low-cost SoGP evaluations, and would not rely on the original solvers, the stratified MC is known to be inefficient in this situation [130]. For computational efficiency, we propose to use an MC sampling strategy inspired by [131]. In view of (4.57), the computation of the $V_i(x_0)$ amounts to the computation of expected value of the squared conditional expectations $\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i}(x_0)\right]^2$,called $E_{1\mapsto i}(x_0)$ hereafter. The expectation $E_{1\mapsto i}(x_0)$ can be rewritten as:

$$E_{1\mapsto i}(x_0) = \mathbb{E}\left[\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i}(x_0)\right]\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i}(x_0)\right]\right], \tag{4.58}$$

leading to the (unbiased) MC estimate

$$E_{1\mapsto i}(x_0) \approx \frac{1}{M}\sum_{j=1}^{M} Y_j Y_j', \tag{4.59}$$

where Y_j, Y_j' are two independent random samples of $\mathrm{G}_{i+1\mapsto n}\circ X_j$ where X_j is a random sample of $\mathrm{G}_{1\mapsto i}(x_0)$. The estimation of the $E_{1\mapsto i}$ for given $x_0 \in \Omega_0$ can be performed in parallel for different GP model i, and can eventually reuse samples X_j from one level i to another. Using the same number M of samples for all the $E_{1\mapsto i}$, and recycling samples, the estimation of the n partial variance V_i has a computational cost of the order of $\mathcal{O}(Mn)$.

As we shall see later, the training strategy may require the evaluations of the partial variances $V_i(x_0)$ at multiple input points $x_0 \in \Omega_0$. In this case, the estimator in (4.59) may be too expensive, in particular if the variance of $\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i}(x_0)\right]$ is large and high accuracy on $E_{1\mapsto i}(x_0)$ is demanded. We then propose in the following two approximations of $E_{1\mapsto i}$ aiming at reducing the computational cost of computing the V_i.

Composition of averages

Following the discussion of Section 4.3, the expected value of composed GP models can be substituted with the composition of the averaged GP predictions. Specifically, we propose to use the following approximation of the conditional average,

$$\mathbb{E}\left[\mathrm{G}_{1\mapsto n} \mid \mathrm{G}_{1\mapsto i}(x_0)\right] \approx \mu_{i+1\mapsto n}\circ \mathrm{G}_{1\mapsto i}(x_0), \tag{4.60}$$

where we have consistently denoted $\mu_{j\mapsto i} := \mu_i \circ \cdots \circ \mu_j$. Using this approximation in (4.57), the contribution of G_i to the total variance is approximated through

$$V_i(x_0) \approx \widehat{V}_i(x_0) = \mathbb{E}\left[\left(\mu_{i+1\mapsto n}\circ \mathrm{G}_{1\mapsto i}(x_0)\right)^2 - \left(\mu_{i\mapsto n}\circ \mathrm{G}_{1\mapsto i-1}(x_0)\right)^2\right]. \tag{4.61}$$

Finally, letting $\widehat{E}_{1\mapsto i}(x_0) = \mathbb{E}\left[\left(\mu_{i+1\mapsto n}\circ \mathrm{G}_{1\mapsto i}(x_0)\right)^2\right]$, we use the MC estimate

$$\widehat{E}_{1\mapsto i}(x_0) \approx \frac{1}{M}\sum_{j=1}^{M}(Y_j)^2, \tag{4.62}$$

where Y_j are independent random samples of $\mu_{i+1\mapsto n} \circ \mathrm{G}_{1\mapsto i}(x_0)$. Compared to the previous estimator, in (4.59), the composition of averages still calls for a full sampling of the whole SoGP chain to get all the V_i at given x_0. Relying on the composition of averages ($\mu_{i\mapsto n}$) instead of the composition of GP processes ($\mathrm{G}_{i\mapsto n}$) reduces the computational cost by reducing the number of random numbers to be generated and also by reducing, to some extent, the variance of the estimator with possibly a lower sampling error in the MC estimate for fixed M.

Linearized approximation

The MC estimation of $\widehat{V}_i(x_0)$ is still random and the sampling noise can cause problems when solving for x_0 the optimization problems associated to the training strategies introduced in Section 4.5. These optimization problems are non-convex and their resolution requires a large number of accurate evaluations of the $\widehat{V}_i$ at multiple x_0. This fact has motivated the second approximation of $V_i(x_0)$ that is both fast to estimate and free of sampling noise.

Starting from the expression of $V_{1\mapsto i}$ in (4.55), we first use (4.60) to obtain

$$V_{1\mapsto i}(x_0) \approx \mathbb{V}\left[\mu_{i+1\mapsto n} \circ \mathrm{G}_{1\mapsto i}(x_0)\right]. \tag{4.63}$$

Relying on a local linearization, we have

$$\mathbb{V}\left[\mu_{i+1\mapsto n} \circ \mathrm{G}_{1\mapsto i}(x_0)\right] \approx \left(\mu'_{i+1\mapsto n}(\mu_{1\mapsto i}(x_0))\right)^2 \mathbb{V}\left[\mathrm{G}_{1\mapsto i}(x_0)\right]. \tag{4.64}$$

The first order derivative of the composition of averages, $\mu'_{i+1\mapsto n}$, can be computed by chain rule differentiation or more generally by finite difference formula. In addition, it is noted that this derivative is considered at the composition of averages $\mu_{1\mapsto i}(x_0)$ rather than at $\mathbb{E}\left[\mathrm{G}_{1\mapsto i}(x_0)\right]$ in order to avoid having to estimate the average of the composition. At this point, Eq. (4.64) provides an approximation of $V_{1\mapsto i}(x_0)$, which, we recall, characterizes the variance induced by the GP models up to G_i. To single-out the effect of G_i and approximate $V_i(x_0)$, we finally consider

$$V_i(x_0) \approx \widetilde{V}_i(x_0) = \left(\mu'_{i+1\mapsto n}(\mu_{1\mapsto i}(x_0))\right)^2 \sigma_i^2\left(\mu_{1\mapsto i-1}(x_0)\right), \tag{4.65}$$

where it is recalled that σ_i^2 is the predictive variance of G_i. By definition, this definition is deterministic and does not call for any MC computations. Moreover, the approximation $\widetilde{V}_i(x_0)$ will be accurate provided that the prediction variances σ_i^2 are small. However, we stress that the estimate will be used to select new training points and Gaussian models to be improved and from this perspective, it needs not be necessarily very accurate.

4.4.3 Generalization

To close this section, we discuss the generalization of the proposed predictive variance decomposition and its approximations above, in the case of more complex SoGP. First, the MC estimates of $V_i(x_0)$ and $\widehat{V}_i(x_0)$ can be readily extended to the case of chained vector-valued SoGP, provided that the final prediction remains scalar, that is $\mathrm{G}_{1\mapsto n}(x_0) \in \mathbb{R}$. The linearized approximation $\widetilde{V}_i(x_0)$ can also be extended to more general chained SoGP, albeit the introduction of the gradient of $\mu_{i+1\mapsto n}$ and the covariance matrix Σ_i^2 of the predictions of G_i.

In addition, one can extend the previous concepts of predictive variance decomposition to more generic SoGP, that is not simply chained one, provided that it remains directed. Specifically, $V_{1\mapsto i}$ becomes the variance of the expected final prediction conditioned all GP predictions upstream of and including G_i, instead of $\mathrm{G}_{1\mapsto i}$. Identically, the composition of averages and the linearized approximations can be derived for more general directed SoGP by substituting the GP predictions G_j downstream of G_i with their averaged prediction μ_j. Note that introducing a tree

representation of the SoGP (and SoS) may help to automate the set of GP models appearing in the conditioning of the variance contribution of a specific GP model. An example of a not simply chained SoS is provided in the result section below.

4.5 Training strategies

In this section, we discuss several strategies for the training of SoGPs. These adaptive training strategies are based on the classical Maximum Mean Square Predictive Error (MMSPE) reduction, which is extended to the SoGP case.

4.5.1 LHS training for SoGP

For simplicity, we consider as previously the case of n simply chained solvers, with real scalar inputs and outputs, and global input x_0 uniformly distributed in a bounded domain $\Omega_0 \subset \mathbb{R}$. We denote $\mathcal{X}_0 = \{x_0^{(l)} \in \Omega_0, l = 1, \ldots, m\}$ a uniform sample set of m global input points; for $i = 1, \ldots, n$ let $\mathcal{X}_i \doteq f_i(\mathcal{X}_{i-1})$ be the images of $\mathcal{X}_{i-1}$ by the solver f_i, such that reusing the notation of the previous section

$$x_i^{(l)} = f_{1 \mapsto i}(x_0^{(l)}), \quad i = 1, \ldots, n.$$

The Gaussian Process G_i, approximating $f_i : \Omega_{i-1} \mapsto \Omega_i$, can be constructed using the sample sets $\mathcal{X}_{i-1}$ and its image $\mathcal{X}_i$ by f_i. Space-filling techniques, such as Latin Hypercube Sampling (LHS) [157] and Sobol sequence [201], can be used to generate the driving sample set $\mathcal{X}_0$, with very satisfying results [202]. The direct application of LHS on Ω_0 will serve as a reference to be contrasted with our sampling strategies proposed below. One advantage of considering sample sets $\mathcal{X}_i$ that are the successive images $\mathcal{X}_0$, is that they implicitly follow the input distribution induced by $f_{1 \mapsto i}$, without having to estimate the distribution of $x_i \in \Omega_i$. As a result, different regions of Ω_i are sampled with a density of training points that reflects their importance. This is a desirable property as it will enforce higher accuracy for the G_i in the regions where they are likely to be queried.

4.5.2 Clustering based training for SoGP

The clustering based approaches presented in Section 4.1.5 are well suited for generating a priori training set in the $\Omega_1, \ldots \Omega_n$ that may have arbitrary boundary and shape. Moreover the joint distribution of the intermediate variables, denoted X_i, is not necessarily independent. Therefore, clustering based approaches such as FFF are well suited for SoGP training set generation. The main difficulty when applying clustering based methods is that the distortion defined in Eq. (4.34) is generally approximated with a MC estimator :

$$D(\mathcal{X}_i) = \frac{1}{N} \sum_{j=1}^{N} \min_{x^l \in \mathcal{X}_i} \left\| x_{i,j} - x^l \right\|, \tag{4.66}$$

and hence a large number of samples $(x_{i,j})_{j<N}$ from X_i have to be generated. Except for $i = 0$ sampling from X_i requires evaluating all the solvers from f_0 to f_{i-1} for a prohibitive computational cost. Instead, we propose a sequential construction of the training sets $\mathcal{X}_0, \ldots, \mathcal{X}_{n-1}$. First $\mathcal{X}_0$ is generated using a LHS approach for instance. Based on $\mathcal{X}_0$, G_0 is constructed and a large number of samples of X_0 are propagated using G_0. The generated samples $(\hat{x}_{1,j})_{j<N}$ are approximately

distributed according X_1 depending on the accuracy of G_0. Using the samples $(\hat{x}_{1,j})_{j<N}$, the distortion defined in Eq. (4.34) can be approximated as :

$$\hat{D}(\mathcal{X}_1) = \frac{1}{N}\sum_{j=1}^{N} \min_{x^l \in \mathcal{X}_1} \left\| \hat{x}_{1,j} - x^l \right\|. \tag{4.67}$$

and the optimal training set $\mathcal{X}_1$ minimizing the approximated distortion $\hat{D}(\mathcal{X}_1)$ can be computed using Lloyd's algorithm. The solvers f_1 is evaluated at $\mathcal{X}_1$, and G_1 in trained using $\mathcal{X}_1, f_1(\mathcal{X}_1)$. This procedure is repeated for G_i and $\mathcal{X}_i$ for $i > 1$ where $\mathcal{X}_i$ minimizes the approximated distortion :

$$\hat{D}(\mathcal{X}_i) = \frac{1}{N}\sum_{j=1}^{N} \min_{x^l \in \mathcal{X}_i} \left\| \hat{x}_{i-1,j} - x^l \right\|, \tag{4.68}$$

and $(\hat{x}_{i-1,j})_{j<N}$ is the image of $(\hat{x}_{i-2,j})_{j<N}$ by G_{i-1}. The procedure is summarized in algorithm 1.

Contrary to the previous section, the $\mathcal{X}_0, \ldots, \mathcal{X}_{n-1}$ are not images of each others. The main drawback of this clustering strategy is that the quality of the training set $\mathcal{X}_i$ depends on the accuracy of G_{i-1}. Hence it is preferable to start with training sample sets large enough so that the constructed GPs accurately reproduces the distribution for the intermediate variables. The errors in the sample sets $(\hat{x}_{1,j})_{j<N}$ are expected to increase with i since they are image of each other through $\mathrm{G}_0, \ldots \mathrm{G}_i$. Moreover, the training sample set minimizing the sample set distortion may not be in Ω_i if for instance Ω_i is not convex [163].

Algorithm 1 Clustering based Training for SoGP

procedure ClusteringTraining($\mathcal{X}_0$, $(x_{0,j})_{j<N}$)
  for $i = 0, \ldots, n-1$ **do**
    Train G_i with $\mathcal{X}_i$ and $f_i(\mathcal{X}_i)$
    for $j = 1, \ldots, N$ **do**
      $\hat{x}_{i+1,j} = \mu_i(\hat{x}_{i,j})$
    Find optimal $\mathcal{X}_{i+1}$ minimizing the approximated distortion in Eq. (4.68)
  return $\{\mathcal{X}_{i=0,\ldots,n-1}\}$

The design of experiment approaches presented in the last two section are completely *a priori* and may not be optimal, in particular for limited size sample set, with a dominant error in regions that have not been sampled. In other words, an adaptive sampling of Ω_0 can yield an error lower than for an a priori LHS and for the same computational complexity measured by the size of the samples set. In addition, adapting the input training set $\mathcal{X}_i$ to each Gaussian Processes appears as a possible way to reduce the error while minimizing the computational complexity, possibly by adapting the size of the samples sets associated to the construction of the different GP models.

4.5.3 Maximum mean square prediction error for SoGP

Given the samples sets $\mathcal{X}_i$, possibly not image of each others and having different sizes, of points $x_i^{(l)} \in \Omega_i$, we denote

$$Q(\mathcal{X}_0, \ldots, \mathcal{X}_{n-1}) = \max_{x_0 \in \Omega_0} \mathbb{V}\left[\mathrm{G}_{1\mapsto n}(x_0)\right], \tag{4.69}$$

where the GP model G_i is constructed using the training sets $\mathcal{X}_{i-1}$ *and its image* $f_i(\mathcal{X}_{i-1})$. In words, Q measures the maximum of the global prediction variance for $x_0 \in \Omega$, given the input

samples sets of each GP model. The prediction variance is classically assumed to be representative of the model error $f_i - \mathrm{G}_i$. Further, the selection of the samples sets $\mathcal{X}_i$ in order to minimize Q is known in the literature as the Maximum Mean Square Prediction Error (MMSPE) criterion [171] or the minimization of the Mean Square Error (MSE) of the Best Linear Predictor [119]. Clearly, computing the $(\mathcal{X}_i)_{0\geq i\geq n-1}$ such that $Q(\mathcal{X}_0, \ldots, \mathcal{X}_{n-1})$ is minimal is a very difficult task even for fixed samples sets size, search over finite sets of candidates $x_i^{(l)}$ [203], or even reducing the search space in Ω_0 and imposing the samples sets to be the images of one to another.

4.5.4 Adaptive training strategies

Adaptive training strategies (or active learning methods) intend to approach the solution of the optimal sampling problem in a greedy fashion, by progressively enriching the samples sets $\mathcal{X}_i$. They offer a cheaper greedy heuristic solution to a complex optimization problem. In the present work, we do not aim to develop an original adaptive method, but rather to propose adaptations of the MMSPE criterion-based method of [171] to systems of GP models. In the following, we propose three different strategies. They are implemented and compared in the section 4.6. As a side note, we remark that the prediction of a SoGP is non-Gaussian therefore, in general, the minimization of the MMSPE criterion is not equivalent to entropy minimization.

4.5.4.1 Global Composition Criterion (GCC)

Following a greedy approach, we propose to select the input point $\tilde{x}_0 \in \Omega_0$ presenting the highest global predictive variance selected to seed the enrichment of the training sets. Specifically, the seeding point is defined as

$$\tilde{x}_0 := \arg\max_{x\in\Omega_0} \mathbb{V}\left[\mathrm{G}_{1\mapsto n}(x)\right]. \tag{4.70}$$

We set $\mathcal{X}_0 \leftarrow \mathcal{X}_0 \cup \{\tilde{x}_0\}$ and update the other input samples sets $\mathcal{X}_i \leftarrow \mathcal{X}_i \cup \{\mu_{1\mapsto i}(\tilde{x}_0)\}$, that is using the successive composed averaged predictions applied to $\tilde{x}$. The update of the training sets for the GCC strategy is outlined in Algorithm 2. In view of (4.70), we call the Global Composition Criterion (GCC) this training strategy.

Algorithm 2 GCC: selection of new training points using the Global Composition Criterion.

1: **procedure** SELECTGCC($\{\mathrm{G}_{i=1,\ldots,n}\}$)
2: Find $\tilde{x}_0 \in \Omega_0$ solution of (4.70)
3: **for** $i = 0, \ldots, n-1$ **do**
4: $\tilde{x}_{i+1} = \mu_i(\tilde{x}_i)$
5: **return** $\{\tilde{x}_{i=0,\ldots,n-1}\}$

Once the training sets $\mathcal{X}_i$ have been enriched, one can proceed with the update of the GP models G_i using $\mathcal{X}_{i-1}$ and its image by solver f_i. The computational complexity of GCC thus amounts to one resolution of *all* the solvers in the system, for every new training point seed $\tilde{x}$. Note that defining the new training points by successive compositions of averages allows parallelizing the computation of their images by f_i. Defining instead $\tilde{x}_{i+1} = f_i(\tilde{x}_i)$ at line 4 of Algorithm 2 would result in a sequential update procedure as f_i must be solved before proceeding with its composition with the next solver f_{i+1}. Furthermore, numerical tests have shown that using the composition of exact images by f_i has no significant impact on the efficiency of GCC.

4.5.4.2 Local Contributions Criteria (LCC)

The strategy GCC is a direct adaption of the classical MMSPE criterion used to train individual GP models. It misses the chain structure of the SoGP, which can incorporate solvers with very different complexity and influence on the on the global output. As a result, areas of the GP models with large predictions errors may not necessarily correspond to composed images of a single area Ω_0. Therefore, we propose here to select for each GP model G_i the input point $\tilde{x}_{i-1} \in \Omega_0$ yielding the highest contribution $V_i(\tilde{x}_{i-1})$. We recall that V_i is the contribution to the global predictive variance of GP model G_i (see Section 4.4.1 and Eq. (4.56)). Specifically, we consider

$$\tilde{x}_{i-1} = \arg\max_{x \in \Omega_0} V_i(x), \quad i = 1, \ldots, n. \tag{4.71}$$

Note that the search space for the $\tilde{x}_i$ is *always* Ω_0. In the following, we call LCC the training strategy based on Local Contribution Criteria in (4.71). Once the input points $\tilde{x}_i \in \Omega$ have been determined, we enrich the respective training sets through $\mathcal{X}_i \cup \{\mu_{1\mapsto i}(\tilde{x}_i)\}$ as underlined by the procedure reported in Algorithm 3. Note that the training point added to $\mathcal{X}_i$ is determined by a composition of averages, $\mu_{1\mapsto i}(\tilde{x}_i)$, and not using the composition of models, $f_{1\mapsto i}(\tilde{x}_i)$, for computational complexity reduction purposes.

Algorithm 3 LCC: selection of new training points using Local Contribution Criteria.

1: **procedure** SELECTLCC($\{\mathrm{G}_{i=1,\ldots,n}\}$)
2: **for** $i = 1, \ldots, n$ **do**
3: Find $\tilde{x}_* \in \Omega_0$ solution of (4.71)
4: $\tilde{x}_{i-1} = \mu_{1\mapsto i-1}(\tilde{x}_*)$
5: **return** $\{\tilde{x}_{i=0,\ldots,n-1}\}$

Because the initial seed $\tilde{x}_i$ is changing from a GP model to another, the LCC strategy does not generate enrichment points that are composed images of one another (neither by f_i or μ_i), with potentially a better reduction of the MMSPE criterion compared to GCC. Comparing further GCC and LCC, updating the GP models for the two strategies has the same computational cost that is one evaluation of every solver in the SoS to compute the image by f_i of the new point added to $\mathcal{X}_{i-1}$. Finally, note that each seed $\tilde{x}_i$ in LCC calls for the resolution of a distinct optimization problem (4.71), whose complexity is comparable to the *unique* optimization problem (4.70) of GCC. However, these optimization problems in (4.71) can be carried out in parallel.

4.5.4.3 Single Model Selection (SMS)

The GCC and LCC strategies require the evaluation of *all* the solvers and update all the GP models. In practice, especially for limited size sample sets (*e.g.* at the start of the adaptive procedure), the MMSPE $\mathbb{V}[\mathrm{G}_{1\mapsto n}]$ can be dominated by the contributions V_i of few GP processes. In this situation, enriching a single training set $\mathcal{X}_i$, or less aggressively just a few of them may constitute a more efficient strategy to focus the computational resources on the improvement of selected GP models, disregarding the update of relatively more accurate ones. To this end, we propose a strategy called Single Model Selection (SMS), which selects a pair of one seed point and index of the GP the featuring the largest contribution to global predictive variance. The pair solves the following optimization problem

$$(\tilde{x}, \tilde{l}) = \arg \max_{\substack{x \in \Omega_0 \\ i \in \{1,\ldots,n\}}} V_i(x). \tag{4.72}$$

Note again that the optimal point is sought in the global input domain Ω_0, so we have to propagate it to define the new training point of the selected model $\mathrm{G}_{\tilde{l}}$ to be improved. As for the other strategy, we enrich the input training set (and its image by $f_{\tilde{l}}$) through $\mathcal{X}_{\tilde{l}-1} \cup \{\tilde{x}_{\tilde{l}-1}\}$, where the new training point is obtained by the composition of averages $\tilde{x}_{\tilde{l}-1} := \mu_{1\mapsto\tilde{l}-1}(\tilde{x})$. As a result of the selection of a single model to be updated, the SMS strategy has training sets $\mathcal{X}_i$ with variable sizes and, in contrast to the other strategies, only the selected GP model $\mathrm{G}_{\tilde{l}}$ needs be updated in SMS. Besides, because of possible large heterogeneities between solvers, it may be interesting to account for the computational cost of solving f_i when selecting the new training point. To this end we extend the optimization problem (4.72) to

$$(\tilde{x}, \tilde{l}) = \arg \max_{\substack{x\in\Omega_0 \\ i\in\{1,\dots,n\}}} V_i(x) - \alpha C_i, \tag{4.73}$$

where C_i is an estimate of the computational cost of solver i and $\alpha > 0$ a user defined constant. The procedure for selecting points in SMS is outlined in Algorithm 4.

Algorithm 4 SMS: selection of a unique new training point by the Single Model Selection.

1: **procedure** SELECTSMS($\{\mathrm{G}_{i=1,\dots,n}\}$,$\alpha$)
2: Find couple $(\tilde{x}, \tilde{l})$ solution of (4.73)
3: $x_{\tilde{l}-1} = \mu_{1\mapsto\tilde{l}-1}(\tilde{x})$
4: **return** $(x_{\tilde{l}-1}, \tilde{l})$

We observe that the strategies presented above rely on training sets that are images of one to another by the true solvers, at least partially (for the members of the initial sets). Ideally, one would like to construct the surrogates without having to perform any computation of the full SoS. A fully decoupled construction of the GP models associated to each solver would be possible if their inputs range and distribution were known a priori. Without this knowledge, one can instead adopt a sequential construction method where all upstream GP models are constructed with sufficiently high precision to ensure a correct prediction of the outputs distribution by the composition of averages. This distribution can then be substituted to the unknown distribution of the inputs of the downstream solvers to proceed (sequentially) with the construct of their GP models. Our estimates of prediction variance and error control strategy SMS, applied to the prediction of the upstream solvers' outputs, can be employed to that end. However, this sequential construction is likely to result in a sub-optimal strategy, as it may consume resources to obtain accurate intermediate GP models that have a weak impact on the prediction variance of the terminal solver. Still, the sequential and fully decoupled approach just described may present an interest when assembling the whole SoS is not possible: it can be used to generate an initial coarse SoGP, which can be refined subsequently using one of our active learning strategies.

4.5.5 Training algorithm

The three strategies presented above are greedy and add a single training point per samples set $\mathcal{X}_i$ (in all sets for GCC and LCC, in a single set for SMS) before updating the GP model(s). For parallelization purposes, one may be interested in adding a batch of training points instead of a single one, in order to run, in parallel, multiple evaluations of the model f_i. In the following, we denote N_{add} the number of training points added at a time to reduce the prediction error of the SoGP. The main difficulty in adding N_{add} points at once is that the three strategies above rely on optimization problems that will produce the same new training points unless the SoGP is

updated. In other words, these strategies are sequential, and it is necessary to update the SoGP for each new training point to obtain the next one.

We propose here to circumvent the sequential nature of the strategies by substituting the evaluations of the models f_i, at the new training points, with the current best predictions of the corresponding G_i, that is using μ_i in place of f_i. Doing so, one does not change the prediction but locally reduces (in the neighborhood of the new training point) the predictive variance σ_i. This reduction, in turn, affects the global predictive variance and its decomposition, such that the next training points will be found at different locations. Algorithm 5 outlines the procedure for adding a batch of N_{add} new training points. The procedure uses one of the three selection procedures (see line 5) to construct the enrichment $\tilde{\mathcal{X}}_i$ of the initial training sets, while updating the GP models G_i (see line 7) using the initial training points with their images by f_i and the enrichment points and their prediction with μ_i. Note that in the case of the SMS strategy, only $\mathrm{G}_{\tilde{l}}$ needs be updated and that one can keep the hyper-parameters of the GP models constant during these updates to further reduce the computational load. The procedure in Algorithm 5 returns the enriched sets to train each model. To this end, the images by the solvers f_i of the N_{add} new training points must be computed first, possibly in parallel, as sought by the approach. Then, the GP models can be recomputed with the exact images and selection of the hyper-parameters.

Algorithm 5 Training algorithm.

1: **procedure** SELECTBATCH($\{\mathrm{G}_{i=1,\dots,n}\}$, $\{\mathcal{X}_{i=0,\dots,n-1}\}$, N_{add}, [,α])
2: **for** $i = 0, \dots, n-1$ **do**
3: $\tilde{\mathcal{X}}_i = \emptyset$
4: **for** $p = 0, \dots, N_{\text{add}}$ **do**
5: $\{\tilde{\mathcal{X}}_{i=0,\dots,n}\} \leftarrow \{\tilde{\mathcal{X}}_{i=0,\dots,n}\} \cup$ SelectStrategy($\{\mathrm{G}_{i=1,\dots,n}\}$[, α]))
6: **for** $i = 1, \dots, n$ **do**
7: Update G_i using $(\mathcal{X}_{i-1}, f_i(\mathcal{X}_{i-1}))$ and $(\tilde{\mathcal{X}}_{i-1}, \mu_i(\tilde{\mathcal{X}}_{i-1}))$
8: **return** $\{(\mathcal{X}_i \cup \tilde{\mathcal{X}}_i)_{i=0,\dots,n}\}$

As a final note, we observe that the optimization problems in (4.70)-(4.73) are nonconvex, in general, and present many local optima as illustrated in see Section 4.6.2. Furthermore, depending on the approximation of global predictive variance or contributions V_i considered, only noisy evaluations of the objective functions may be available, such that appropriate optimization procedures must be considered. Fortunately, the precise computation of the optimal points is not critical to the efficiency of the training procedure, in particular when using a batch of points. In the present work, we relied on genetic algorithms [204] to approximate the solutions of the optimization problems (4.70)-(4.73).

4.6 Test problems

The proposed SoGP methodology is now applied on several test cases corresponding to different types of SoS, in terms of dependencies and structure. It is then applied to a realistic engineering system of solvers in the context of the space object reentry.

The first test case is a simple SoS with two chained solvers each having a single input. The second test case consists of four solvers, with eight global inputs. The first three solvers are independent and have a single output constituting the inputs of the last solver. The third test case is composed of four solvers directly chained where a solver has for inputs the output

of its upstream solver plus some global inputs (overall 16 global inputs). This structure is representative of many systems of solvers used in industry.

In order to assess the accuracy and robustness of the proposed framework, a systematic comparison is performed with a global GP constructed on the whole SoS considered as a black box.

In particular, we compare the following methods:

- *BB-LHS*: a global GP is built on the whole SoS considered as a black-box, using a LHS-based sampling;
- *BB-MMSPE*: a global GP is built on the whole SoS considered as a black-box, using the MMSPE training strategy;
- *SoGP-LHS*: a SoGP is built, using a LHS-based sampling;
- *SoGP-clustering*: a SoGP is built, using a clustering based training set (described in Section 4.5.2) ;
- *GCC*, *LCC* or *SMS*: a SoGP is built, using the GCC, LCC or SMS training strategy (described in Section 4.5), respectively.

In all cases, the performance of the method is evaluated by computing a normalized L_2-error norm on the global output approximation. Denoting y the exact SoS global output and $\widetilde{y}$ its approximation (using one of the proposed methods), the error is estimated using N independent Monte Carlo samples of the global inputs, as follows:

$$Err_{L_2}^2 \approx \frac{\sum_{i=1}^{N} \left(\widetilde{y}(x_i) - y(x_i)\right)^2}{\sum_{i=1}^{N} y(x_i)^2}, \tag{4.74}$$

where the $x_{1 \leq i \leq N}$ are independent Monte Carlo samples of the global inputs.

For each test case and method, the SoGP are initialized using an initial LHS set in the global input space. In the case of the GCC, LCC, and SMS adaptive strategies, this initial LHS set is progressively enriched adding a new batch of N_{add} training points following the strategy discussed in Section 4.5 (see also Algorithm 5). Also, to assess the influence of the random generation of the initial LHS sample, the numerical experiments are repeated several times. We report the errors by their values averaged over the repetitions, along with lower and upper bounds corresponding to the best and the worst errors over the set of repetitions. Since the number of repetitions never exceeds 10, the errors' bounds shown are not precise estimates of the errors statistics, but constitute a rough characterization of the methods' variability. Table 4.1 summarizes the default parameters used in the numerical experiments: the initial LHS sample size, batch size N_{add}, and the number of repetitions used in the three test cases presented below. In all cases, the inputs have independent uniform distributions. We also recall that the training methods do not require the same number of SoS evaluations: the GCC and LCC methods require the evaluation of the all solvers in the SoS, for each new training point, whereas the SMS only requires only the evaluation of the selected solver.

4.6.1 Test case 1

This first test case consists in the composition of the two univariate functions f_1 and f_2 presented in Fig. 4.7. These functions are defined as follows:

$$f_1 : x \mapsto \exp\left(\sqrt{x}\right)\sin(x) + 6\exp\left(-(x-2)^2\right) + \frac{5}{2}\exp\left(-3(x-1)^2\right), \tag{4.75}$$

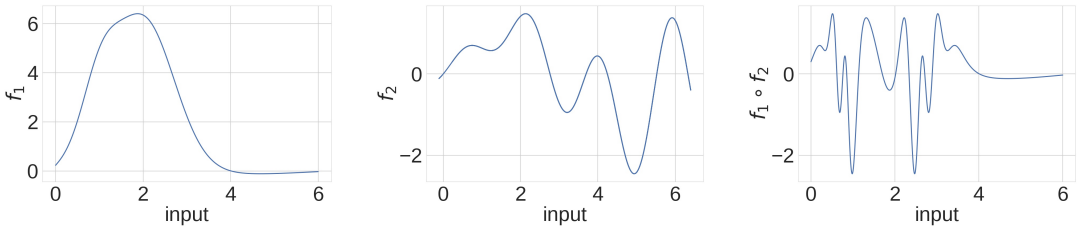
Test case	initial LHS set size	batch size N_{add}	number of repetitions
Test case 1	5	5	10
Test case 2	150	50	5
Test case 3	500	50	5

Table 4.1: Default parameters for the three test cases.

and

$$f_2 : x \mapsto \sin(x) + 0.3 \times x \times \sin(3.4x + 0.5) \tag{4.76}$$

The global output is defined as $f = f_1 \circ f_2$. The global input is uniformly distributed between 0 and 6.



(a) First solver $f_1(x)$ (b) Second solver $f_2(x)$ (c) SoS $f_1 \circ f_2(x)$

Figure 4.7: SoS for the test case 1.

Figure 4.8 reports the results obtained for test case 1. We observe that the SoGP-based approaches systematically achieve a lower error than the global (BB) approaches, by at least one order of magnitude. Several remarks can explain this result. First, the SoGP-based approach is applied to approximate relatively simple functions, f_1 and f_2, whereas the global approach works on a much more complex function $f_1 \circ f_2$, which features a highly multimodal behavior and a plateau. As a consequence, intuitively, we can expect that the approximation of $f_1 \circ f_2$ should be more challenging than approximating f_1 and f_2 solely. In general, as long as the composing functions are simpler than the final output, it is expected that a SoGP-based approach will perform better. Second, in the global approach, a part of the information available in the training set is not used because the evaluations of the first solver are not taken into account. As a general remark, since the SoGP prediction is the composition of multiple GPs, it generally depends on more hyperparameters, compared to the global approaches. Having more hyper-parameters to learn could be detrimental to the computational complexity, but this drawback is compensated by the improved approximation capabilities brought by the extended set of hyper-parameters and the additional information brought by the intermediate variables.

The SoGP-LHS and SoGP-clustering are two model free approaches for SoGP training. In this example the clustering based approach presents a clear advantage compared to the SoGP-LHS as SoGP-clustering performs as well as the AL methods. Since there are only two solvers in this case, the only difference between SoGP-LHS and SoGP-clustering comes from the training set $\mathcal{X}_2$ used to learn f_2, since in both cases $\mathcal{X}_1$ is generated with a LHS. This example shows the clear advantage of a generating training set for each solver instead of propagating an initial LHS. While the training samples in SoGP-LHS follow the input distribution of each solver, the clustering training set offer a better coverage of the input space of f_2. The good performance of SoGP-clustering with respect to SoGP-LHS is directly related to the structure of the SoS. In figure 4.9a are represented the errors induced by the second GP for the SoGP-LHS

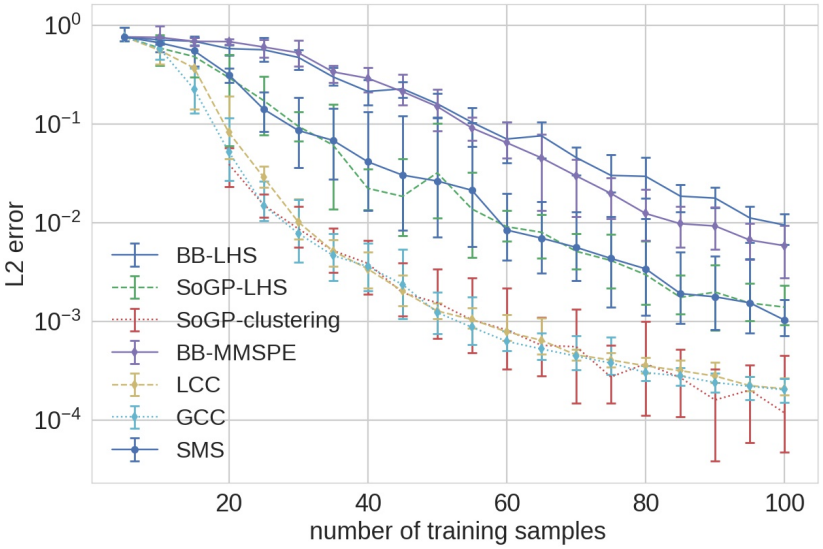
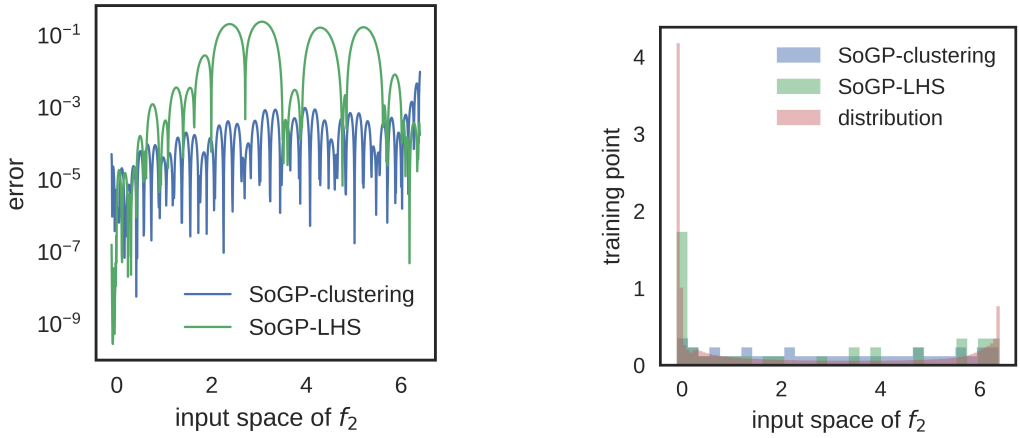


Figure 4.8: L_2 error norm vs the number of training samples for test case 1.



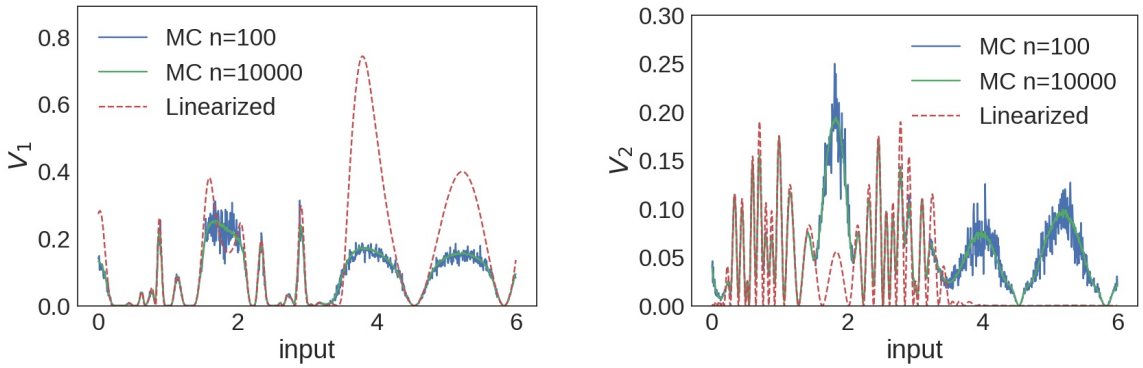
(a) Error distribution induced by the second GP trained with SoGP-LHS or SoGP-clustering training GP

(b) Distributions of the training points for the second GP compared to the second GP input distribution obtained with 40 training samples

Figure 4.9: Comparison of the training set generated by SoGP-LHS and SoGP-clustering

and SoGP-clustering training. The errors are represented in Ω_2, the input space of f_2. The SoGP-clustering globally features a lower error than the SoGP-LHS except in a region near 0. From figure 4.9b, we observe that this region is associated with high probability density. In this case SoGP-LHS over samples high density region and performs locally very well in those regions but lacks a good coverage of the domain.

Differences in performance between the global (BB) and the SoGP-based approaches are even more significant for the adaptive strategies. Every adaptive technique formulated in a SoGP-based framework yields better performances than the MMSPE-BB approach (which is doing slightly better than the standard LHS-BB approach). Regarding the performance of the



(a) First GP model

(b) Second GP model

Figure 4.10: Estimates of the variance contributions V_1 and V_2 of the two GP models, at an initial stage of the construction (involving a total of 25 training points): compared are the linearized approximation (4.65), and the Monte Carlo estimates of (4.62) using 100 and 10,000 samples as indicated.

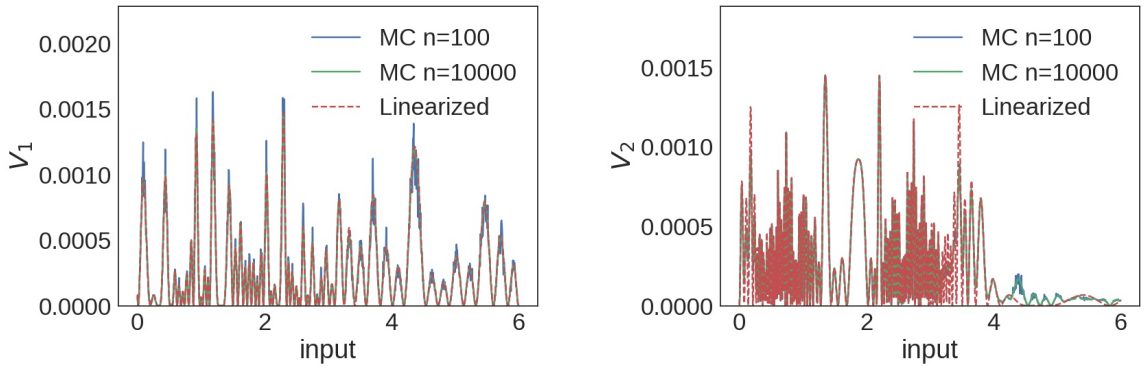
SMS method, we stress that it only requires one solver evaluation per additional training point when the GCC and LCC methods require the evaluation of the whole SoS for each new training point. As a consequence, the results reported in Fig. 4.8 should be cautiously interpreted since the SMS method has a lower computational cost compared to the other methods. The computational cost of the solvers must be considered for a fair comparison. For instance, assuming that the two solvers have the same computational cost, the computational complexity of SMS is half that of the GCC and LCC methods, and the SMS efficiency is comparable to the GCC and LCC efficiencies. In fact, in this example, the SMS method focuses primarily on the second solver and does not add many training points for learning the first solver which is much easier to approximate.

4.6.2 Impact of predictive variance prediction

We take advantage of the simplicity of test case 1 to investigate the impact of different forms proposed in Section 4.4.1 to approximate the contribution V_i of the GP models to the prediction variance. Specifically, we compare the use of the MC estimates V_1 and V_2 given by (4.62) with the approximation given by the linearized form in (4.65).

Figure 4.10 compares the MC estimate using 100 and 10,000 samples, with the corresponding linearized estimate, for the contributions of the first 4.10a and second 4.10b GP models, at an early stage of the construction with 25 training points selected with the criteria of SMS. For the first model, we see that the MC and linearized approximations are in good agreement on most of the input domain. The linearized form, however, is seen to significantly overestimate the contribution to the variance in some area where it is the most significant. Regarding the comparison between the two MC estimates, we remark that they generally agree satisfactorily, except in some localized areas where the sampling noise is significant when only 100 samples are used. Interestingly, the areas of the input space where the sampling noise is noticeable correspond to the areas where the linearized form departs the most from the MC estimates. This finding suggests that the linearized approximation cease to be accurate in places where the variance structure calls for a higher sampling effort. Similar observations hold for the second model, except that the linearized approximation now *underestimates* significantly the MC estimates where it is the most subjected to the sampling noise.

Naturally, one can expect the differences between the MC and the linearized estimates to reduce as the prediction variances of the GP models decrease. Figure 4.11 confirms this expected trend. It compares the previous estimates at a later stage of the construction, involving 85 training points. Note that for a fair comparison, all estimates use the same SoGP construction. Compared to the previous case, the plots confirm that the differences are much less significant; the convergence of the MC estimates also seems to have improved as much smaller differences between the two MC sample set sizes are reported. Besides the lower magnitude and the better agreement between the different estimates, a noticeable evolution between the plots of Figures 4.10 and 4.11 is the sharp increase in the frequency content: the functions V_1 and V_2 are oscillatory with many zero. The complex structure of the contributions to the variance highlights the existence of multiple local maximums that, as mentioned previously, calls for a robust optimization procedure for the selection of the next training points.



(a) First GP model

(b) Second GP model

Figure 4.11: Estimates of the variance contributions V_1 and V_2 of the two GP models, at an initial stage of the construction (involving a total of 85 training points): compared are the linearized approximation (4.65), and the Monte Carlo estimates of (4.62) using 100 and 10,000 samples as indicated.

To complete the analysis of the different estimates of the V_i, we provide in Figure 4.12 a comparison of the resulting global surrogate errors, obtained for the same SMS strategy but based on the different estimates. We see that, on this experiment, the method selected for the estimation of the V_i has only a weak influence on the resulting surrogate errors, with differences that are comparable to the variability bounds of the SMS method shown before in Figure 4.8. We can conclude that the linearized estimate is quite robust, on this example, and should be preferred because of its much lowest computational cost. The MC estimation constitutes a more expensive approach, but it is perhaps safer, especially at the early stage of the construction when the predictive variance is large. Regarding the MC estimate, the presented results suggest that it is unnecessary to use a large number of MC samples. However, an optimization procedure able to deal with noisy evaluations must be employed to determine the next training point. Such procedure usually comes with a higher numerical cost, so that the overall interest of using the MC estimate may be limited in practice.

4.6.3 Test case 2

In this test case, the SoS consists of three independent solvers, with independent global inputs, and which scalar outputs are the inputs of the downstream solver. Overall, the SoS has four solvers and eight global inputs as depicted in Fig. 4.13. As seen from the figure, the SoS of test

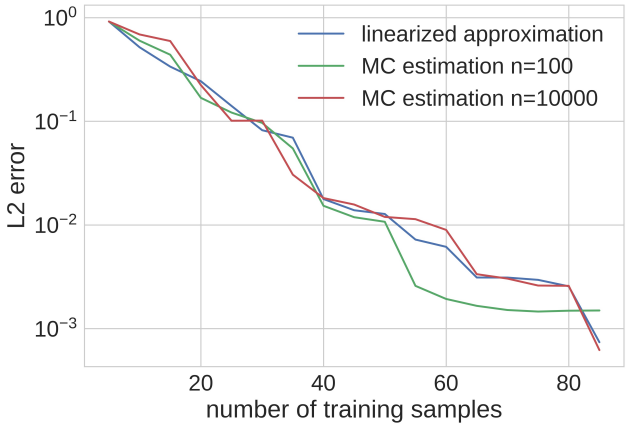


Figure 4.12: L_2 error norm vs. the number of training samples for Test-Case 1 using the SMS criterion with the linearized approximation of the variance decomposition and the Monte Carlo estimation obtained with 100 samples or 10,000 samples. All cases are initialized with the same training set of 5 points.

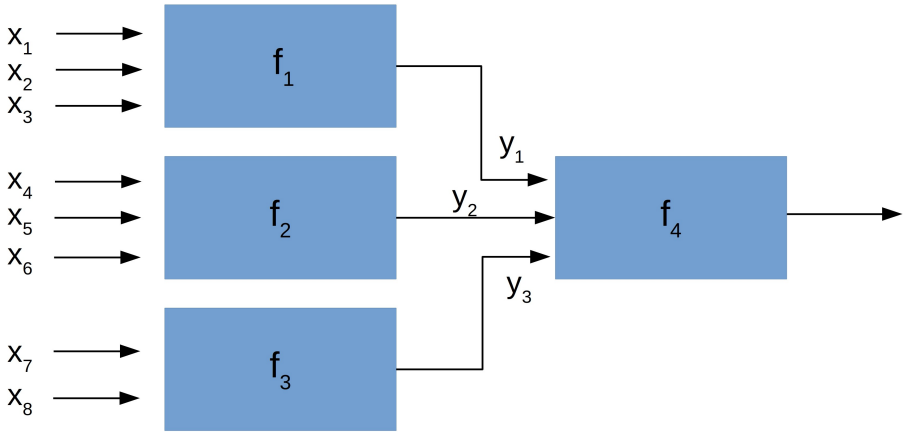


Figure 4.13: SoS for test case 2.

case 2 has two blocks consisting of solvers 1-3 and solver 4, respectively. The individual solvers are defined by the analytical functions below:

$$f_1(x_1, x_2, x_3) = \sin(2x_1x_2) + x_3^2, \quad f_2(x_4, x_5, x_6) = 3x_4^2x_5^2x_6,$$
$$f_3(x_7, x_8) = 2x_7 + x_8^3, \quad f_4(y_1, y_2, y_3) = y_1 + y_1^2 \sin(y_2)\cos(y_3).$$

The global inputs have a uniform and independent distribution between 0 and 1.

The training strategies formulated in Section 4.5 are compared with an additional one. Because of the SoS structure, two options are possible in the SMS strategy: i) to identify and train the most unreliable solver (which is the original SMS technique); ii) to use SMS to identify and train the most unreliable block (solvers 1-3, or solver 4). This last strategy is denoted in the following as *SMS-block*. For this test case, we compare both approaches together. Figure 4.14 reports the errors of the different strategies on this test case. It shows that the SoGP-based approaches outperform the global black-box approaches, with errors reduced by up to two orders of magnitudes. We first observe that for the non adapted methods based on LHS, the simple SoGP method (SoGP-LHS) does better than the global BB method (BB-LHS), with an error

about 3 to 5 times less. The improvement is not as significant than in the previous test case. The SoGP-clustering is the most efficient non adapted method with a error twice smaller than the SoGP-LHS. In this case too, it is advantageous to improve the coverage of the input space of f_4. As for the first test case, the input distribution of f_4 represented in figure 4.15 features high density regions where the SoGP-LHS may be oversampling and may cover poorly less probable regions. While putting more training points in high density regions is intuitively a good feature for a training set, large errors in lower density regions may offset the good performance in high density regions as we observed in this test case and the previous one.

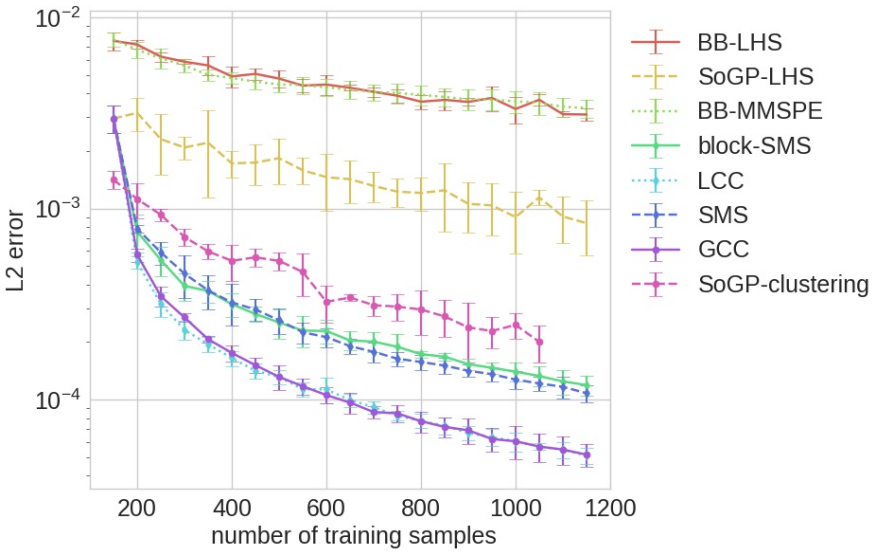


Figure 4.14: L_2 error norm vs the number of training samples for test case 2.

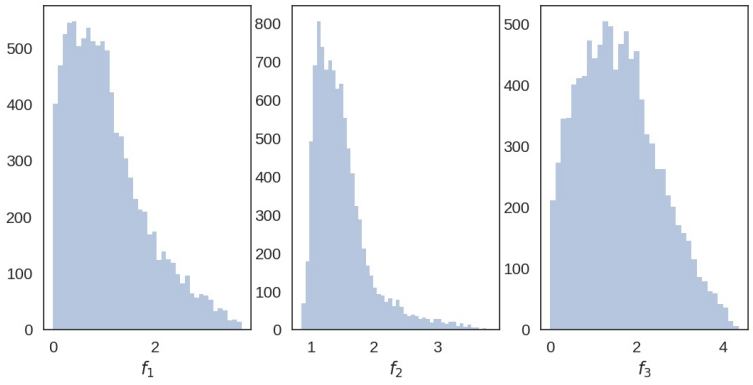


Figure 4.15: Marginal histograms of the outputs of solver 1, 2 and 3 (from left to right, in arbitrary unit).

Focusing on the performances of the training strategies, we observe that the global MMSPE-BB method performs particularly bad, on this test case, since it brings no improvement compared to the non-adapted global BB-LHS approach. This effect is a well-known issue of MMSPE adaptivity in high dimension since, as mentioned in [179, 177], the MMSPE tends to place training points at the edge of the domain. This behavior deteriorates the performance of the

global surrogate model when the inputs dimension is high. This issue appears to be significantly mitigated for the SoGP methods with MMSPE-based adaptive strategies (GCC, LCC, and SMS), owing to the reduced dimensionality of the individual GPs inputs. Compared to the previous test case, the adapted strategies clearly outperform the non adapted SoGP approaches (SoGP-LHS and SoGP-clustering). Concerning the relative performances of the SoGP-based adaptive strategies (SMS, SMS-block, LCC, GCC), GCC and LCC yield similar performances. This result is due to the SoS specific structure in which the last solver (f_4) contributes the most to the global predictive variance. As a consequence, the two methods end up selecting the same enrichment point for the second block. Concerning the SMS-based strategies, SMS and SMS-block perform identically because, again, the last solver is the hardest to learn. Looking to Figure 4.14, where the errors are plotted as functions of the number of training samples, SMS, and SMS-block methods seems to have slightly lower performance than the LCC and GCC method, as this representation does not reveal differences in computational cost. In order to highlight the gain in using SMS-based techniques, which are the most computationally effective techniques for this test case, we report in Fig. 4.16 the L_2 error with respect to the computational cost, computed here as the number of calls to a solver (assuming implicitly that each solver has the same evaluation cost).

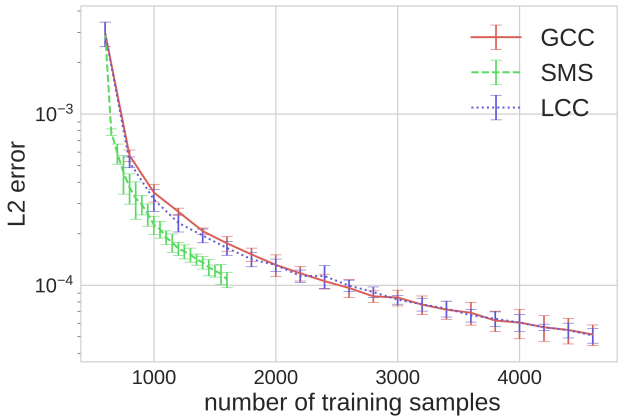


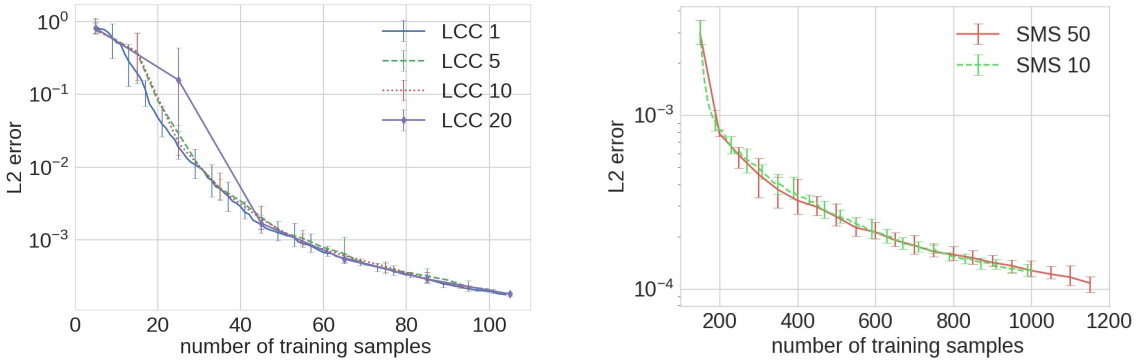
Figure 4.16: L_2 error norm vs the number of solvers evaluations for test case 2.

4.6.4 Influence of the training batch size N_{add}

As discussed previously, the most effective training strategy should consist of adding one training point at a time and run the SoS in a purely sequential manner, without thereby exploiting the full potential of parallel computing. By setting $N_{add} > 1$ in algorithm 5, it is possible to evaluate in parallel a whole set of new training points for the same solver. It remains the question of selecting the batch size N_{add} offering the best trade-off between the cost reduction of the solver evaluation, thanks to parallelism, and a less effective adaptation due to a non-optimal sequential choice of the points. In this section, we investigate the influence of the training batch size on the SoGP performance for the test cases 1 and 2, and using the LCC and SMS strategies respectively, since they were found to be the most effective ones.

Figure 4.17a shows the convergence of the L2 error in test case 1 for different batch sizes $N_{\text{add}} = 1, 5, 10$ and 20, and the LCC method. Each experiment is repeated ten times. In this cases, the asymptotic performance of the adaptive methods appears to be virtually insensitive to N_{add}, indicating that one can take advantage of the parallelism over multiple samples without

affecting the convergence of the method. This conclusion is also valid for test case 2 when using the SMS strategy with $N_{\text{add}} = 10$ and 50, as shown in Fig. 4.17b. This later example also indicates a slightly greater variability of the error with the initial LHS sample set.



(a) Test case 1 with LCC method

(b) Test case 2 with SMS method

Figure 4.17: L_2 error norm2 vs the number of training samples and for different batch size N_{add}. Test case and method as indicated.

In practical applications, one should select N_{add} according to the computational budget and the final number of training samples to be added. It should be as large as possible to exploit parallel solvers evaluations, but not too large, especially in the first stages of the adaptive procedure, to avoid amplifying the initial sample variability.

4.6.5 Test case 3

This test case consists of a SoS with 16 global inputs and four solvers organized as depicted in Fig. 4.18. The first solver is a Sobol function [129] depending on five parameters, defined as follows:

$$f_1(x_1, x_2, x_3, x_4, x_5) = \prod_{k=1}^{5} g_k(x_k), \tag{4.77}$$

where $g_k(x_k) = \frac{|4x_k - 2| + a_k}{1 + a_k}$, $a = (12, 2, 3, 4, 45)$.

The second solver is the Ishigami function [205] defined as:

$$f_2(x_1, x_2, x_3) = \sin(x_1) + a \sin^2(x_2) + b x_3^4 \sin(x_1), \tag{4.78}$$

with $a = 7$ and $b = 0.1$. The input x_1 of solver f_2 is the output of the first solver, *i.e.* f_1. The other two solvers are products of polynomial functions and trigonometric functions, defined as follows:

$$f_3(x_1, x_2, x_3, x_4, x_5, x_6) = x_2^2 \arctan(1 - x_6) + x_3 x_4 x_5^3 + 3x_1, \tag{4.79}$$

where x_1 of f_3 is the output of the second solver, *i.e.* f_2;

$$f_4 = (x_1, x_2, x_3, x_4, x_5) \mapsto \sin(x_5) x_4 + x_1 x_2 + x_3, \tag{4.80}$$

where x_1 of f_4 is the output of the third solver, *i.e.* f_3. The global inputs are uniformly and independently distributed between 0 and 1.

This SoS differs from the previous ones since each solver has some global inputs in addition to the output of the upstream solver. This structure is representative of many industrial SoS, such as the one considered in the following section. Figures 4.19 and 4.20 presents the results

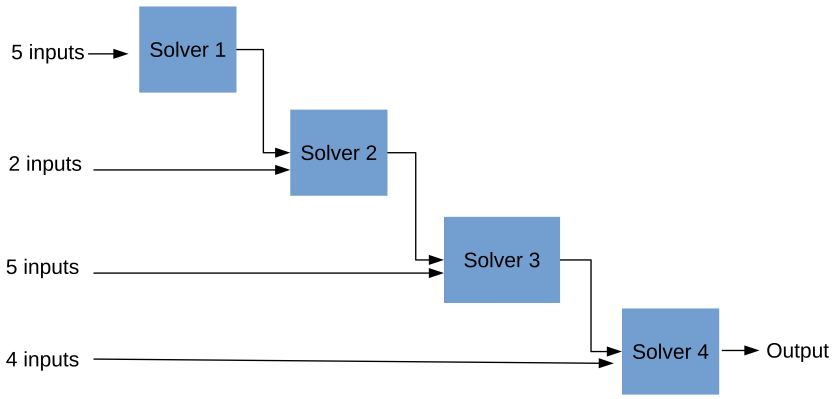


Figure 4.18: Test case 3: cascade-like SoS structure with 16 global inputs and 4 solvers.

for this test case. Focusing on the errors as functions of the sample set size, shown in Fig. 4.19, it is seen once more that the SoGP framework brings significant gain in accuracy over the global black-box approaches (BB-LHS and BB-MMSPE). We again explain this gain by the relatively lower dimensionality of individual solvers inputs, compared to the global SoS. Also, we remark that the individual solvers are complex functions (in particular the Sobol and Ishigami functions) that yield a very complex output when composed together.

In this test case, the accuracy gain of adaptive strategies is less significant than in the previous test cases. In particular, the global BB-MMSPE performs very poorly with a stagnating error after 1500 samples. As before, this behavior is expected in high dimensional problem [177], as in the present test case. In contrast, the GCC, LCC and SMS strategies perform much better than BB-MMSPE, although they have relatively high dimensional inputs. The dimensionality translates into a slow decay of the error with the number of training samples. As in the previous test cases, a fair comparison between SMS, GCC and LCC should consider the lower computational cost of the SMS. This perspective can be appreciated from Fig. 4.20 which depicts the error as a function of the number of calls to a solver, assuming again that all the solvers have the same cost. The SMS is seen to yield the lowest error for a given computational cost. The gain of SMS primarily comes from the identification of the most unreliable solver, here the Sobol function, which is responsible for most of the predictive variance. As a consequence, focusing the computational effort on this solver is very efficient and improves the performance of the whole SoS. This example illustrates the interest in identifying the solver yielding the most of variance in a SoS.

4.6.6 Preliminary test for space simulation

Before applying the framework to a full scale industrial problem, we take an intermediate step and apply the SoGP framework to a "toy" reentry predictor developed by ArianeGroup. This SoS is used to predict the trajectory of a reentering space object and consists of two solvers: i) an upper atmosphere trajectory solver; ii) a three degree-of-freedom trajectory solver integrating a thermal module that provides the temperature of the object [206]. The upper atmosphere solver computes the trajectory from 120 km to 80 km of altitude, and the second solver propagates

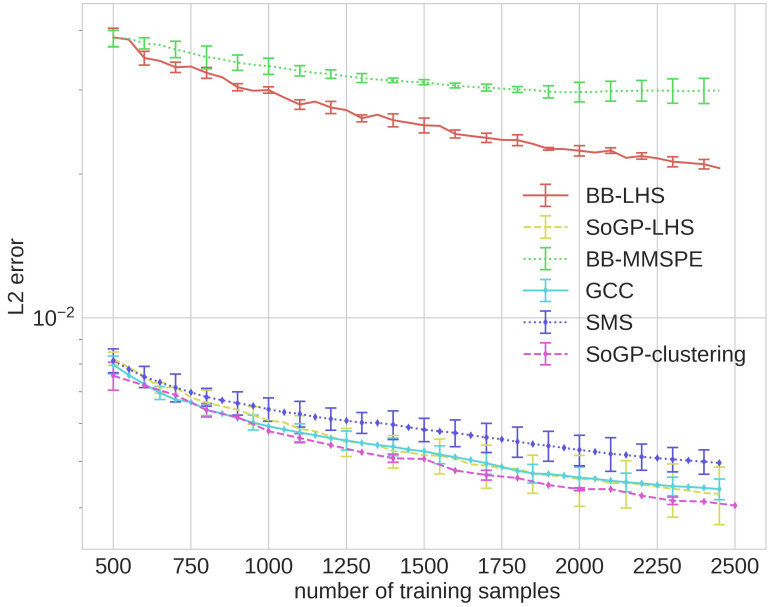


Figure 4.19: Errors *vs* training set size for the Cascade-like SoS

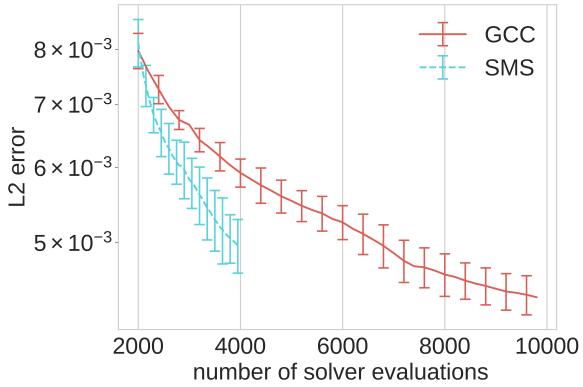


Figure 4.20: Errors *vs* number of calls to solvers for the Cascade-like SoS.

the trajectory from 80 km to 75 km. The output of the first solver is the object position and velocity at an altitude of 80 km, which are the inputs to the second solver. The final quantity of interest is the object temperature at 75 km of altitude, a quantity of interest for the estimation of the breakup risk. We consider a spherical pressure tank as a reentering object.

The structure of the SoS is illustrated in Fig. 4.21, where the global inputs are also depicted. These global inputs are the following: i) 6 inputs for the initial flight conditions (object initial position and velocity); ii) 2 inputs for the atmospheric temperature and density in the upper atmosphere; iii) 3 inputs for the material thermal and emissivity properties. In total, 11 global inputs are considered and modeled as random variables with uniform distributions. In this SoS, the two composing solvers present a very different complexity. The first one is an almost linear mapping between the inputs and the outputs. It is therefore straightforward to learn. On the contrary, the second solver displays a more complex behavior and is so harder to emulate. Finally, we mention that the computational costs of the two solvers are different with an estimated ratio of 43:7 between Solver 1 and Solver 2. At this point, the input uncertainty ranges and

distributions are not of interest as the purpose of this test is to validate the approach on a test case close to the real application.

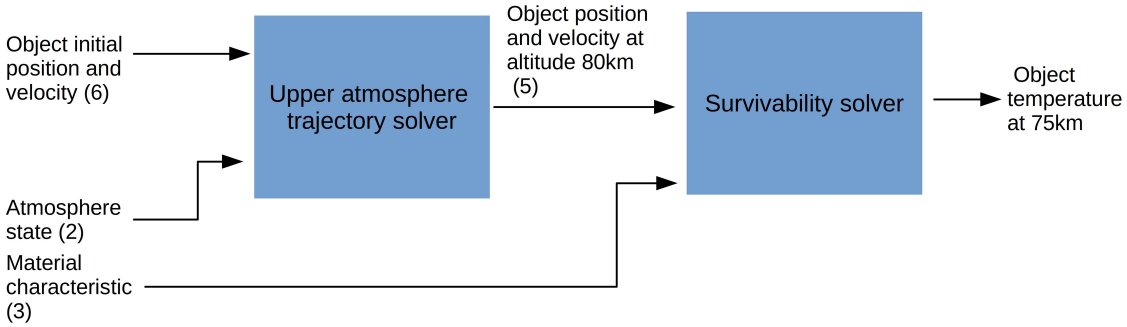


Figure 4.21: SoS for the space object reentry simulation.

Three approaches are contrasted on this problem : the global BB-LHS, the non-adaptive SoGP-LHS, and the adaptive SMS methods. The global BB-LHS and non-adaptive SoGP-LHS methods are applied on four samples LHS sample sets of size 200, 300, 500 and 600 respectively, while the SMS strategy is initialized with the LHS set of dimension 500 before selecting, in batch of size $N_{\text{add}} = 10$ new points, till an equivalent of 600 solver evaluations is reached.

The results are summarized in Fig. 4.22. The figure depicts the L_2 error norm as a function of the computational cost, reported as the number of solver evaluations scaled by their respective relative costs (0.86 and 0.14 for the upstream and downstream solvers, respectively). In these experiments, the error is estimated using an independent set of 1000 LHS points. Consistently with the previous test cases, the SoGP-LHS performs better than the global BB-LHS method. The improvement is explained by the number of inputs of the solvers which is less than the dimensionality of the global inputs. Concerning the adaptive strategy, the SMS approach presents an unusual behavior with an initial dramatic decrease in the error, until a computational cost of around 520 (*i.e.* for a few adaptive batches). Beyond this point, the error decays at a much slower rate, although it remains lower than for the other LHS-based methods. We explain this behavior as follows. Initially, the SMS strategy adds sample points to improve exclusively the second solver, which cost is only 14 % of the whole SoS chain. As a result, the SMS performs exceptionally well during this phase. Subsequently, when the contribution of the two solvers to the overall prediction variance is balanced between the two solvers, the advantage of selecting a particular solver is less critical, and the limits of the MMSPE-based strategy in high dimension [177, 179] become apparent with a stagnating error decay as a result.

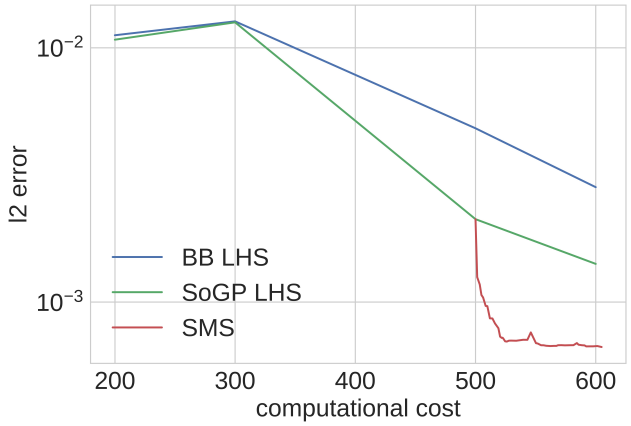


Figure 4.22: L_2 error norm vs the computational cost for the space object reentry SoS.

4.7 Conclusion

In this chapter, we have proposed a framework for constructing a system of Gaussian Processes (SoGP) approximating a directed system of solvers (SoS) having uncertain inputs. The SoGP substitutes GP models to the solvers constituting the SoS. The prediction is then obtained from the composition of the GP models. This contribution provides an efficient tool for propagating uncertainties in the simulator we developed in Chapter 3.

We have demonstrated that this approach is potentially more efficient than the direct construction of a global surrogate of the SoS, in particular when the SoS combines individual solvers with low dimensional inputs and simple mappings from inputs to outputs. In these situations, a reduction of orders of magnitudes in the L_2 error norm can be achieved for the same number of training samples. In addition, the SoGP construction involves an extended set of hyperparameters (typically a set for each model, instead of a unique set for the global mapping from the inputs to the outputs) with improved approximation capabilities as a result.

By design, our proposed SoGP is suited to parsimonious active learning strategies. The active learning strategy is classically based on algorithms requiring a predictive error estimation. In this work, the global predictive variance estimations of the SoGP is used to assess the precision of the prediction. A formal decomposition of the global predictive variance is derived to identify the contribution of each solver. Different approximations of the solver contributions have been proposed to improve the computational efficiency. These estimates have been used to extend the MMSPE-based adaptive algorithm, with improved performance compared to non-adaptive strategies on several test-cases. Specifically, the three training strategies proposed (GCC, LCC, and SMS) have yielded systematically better results, up to one order of magnitude error reduction, compared to the non-adapted SoGP-based approach. In particular, the SMS strategy which selects the specific solver with the highest contribution to the predictive variance is shown to be computationally very effective, in particular when some GP models of the SoGP have a dominant contribution to the prediction variance and low evaluation cost.

Numerical tests also revealed some limitations of the considered active learning strategies which call for improvements in view of applications to engineering problems. These limitations are not related to our SoGP framework but are rather generic to the MMSPE criteria and its lack of robustness, in particular for high-dimensional inputs. Potential improvements of this aspect could involve the extension of the solver (and training point) selection using the *integral* prediction variance reduction criteria, which is known to constitute a more effective

approach [177, 180]. This extension would, however, require additional developments to obtain computable estimates of the solver contributions.

In the next chapter, the SoGP framework is used to efficiently propagate the uncertainties in the reentry model constructed in the previous chapter.

Chapter 5

Uncertainty Analysis of Space Object Breakup

In this chapter, we apply the framework introduced in Chapter 4 to the reentry of an upper stage from its GTO orbit. The object reentry is controlled; that is, the object performs a deorbiting maneuver to leave the GTO orbit and reenters the atmosphere in a targeted region. The objective is to provide a robust estimate of the breakup conditions, to identify the contributions from each uncertainty to the breakup conditions, and to study the statistic of critical quantities such as the object temperature or the heat flux. In this chapter, the survivability of the fragments released after the breakup is not considered. The main contributions of this chapter are :

- *Successful test of the SoGP framework developed in chapter 4 on a full scale industrial application;*
- *Demonstration of the interest of using a probabilistic model to predict the breakup;*
- *Accurate and rigorous propagation of a large number of uncertainties and identification of the most influential uncertainties using sensitivity analysis.*

5.1 Upper stage reentry from GTO orbit

In this section we present the numerical model, the main characteristic of the GTO reentry and how the uncertainties are modelled for this problem.

5.1.1 Numerical model

The simulator developed in chapter 3 is used for the initial stages of the simulation until the object breaks up. The survivability of the fragments is not investigated in this chapter. A graphical representation of the simulator is shown in Fig. 5.1. Recall that in the case of a controlled reentry, the simulation starts with the deorbiting model. The deorbiting model takes as input the initial orbit elements, the deorbiting boost characteristics (time, amplitude and orientation) and returns the object flight conditions at 120 km of altitude. In parallel, the aerodynamic solver (ARPEGE) generates the aerodynamic tables function of the object orientation, Mach number and Knudsen number based on the object geometry represented in Fig. 5.2. Using the aerodynamic tables and the flight conditions at 120 km, the trajectory solver BL43 coupled to the thermal solver ADRYANS V4 is launched. It returns the object position, velocity, temperature and mass evolution over time. This information is directly used by the probabilistic breakup

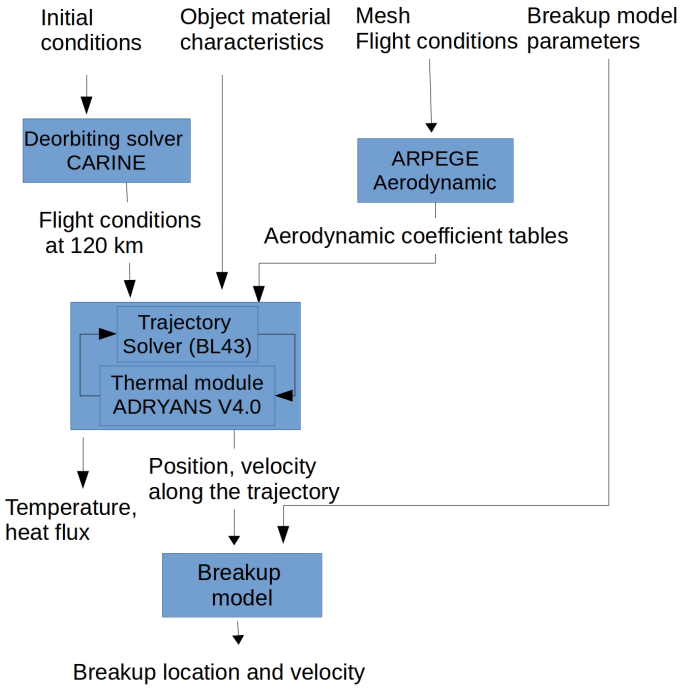


Figure 5.1: Breakup simulator

model to return the breakup probability distribution. The probabilistic model described in section 3.2.2 defines two epochs t_{init} and t_{end} between which breakup can occur uniformly. To define t_{init} and t_{end} an object-oriented approach is chosen. The upper stage structure is composed of a shell that is heated and the internal components that are release during the breakup. The quantities t_{init} and t_{end} depend on the shell temperature and the percentage of ablation.

We compute the flight conditions (position and velocity) at different times of the reentry before and at breakup. We are also interested in the object temperature and the convective heat flux.

5.1.2 Object description

In a GTO controlled reentry, the upper stage is deorbited from its equatorial GTO orbit. The deorbiting maneuver consists of a final backward boost fired to slow down the upper stage and make it fall back on Earth. Fig. 5.2 illustrates the upper stage external structure. It is mostly composed of aluminum and weighs 7000 kg (see Table 5.1). The shell weights 3000 kg and is made of aluminum. Recall that the mode ADRYANS V4 can only consider simplified shapes. In this test, the upper stage is approximated with a cylinder to compute the heat flux. The aerodynamic forces are computed using the full object geometry.

Variable	Value
Internal mass	4000 kg
Shell mass	3000 kg
Material composing the shell	Aluminium
Dimensions	diameter 4.5 m , length (with nozzle) 7 m

Table 5.1: Case description

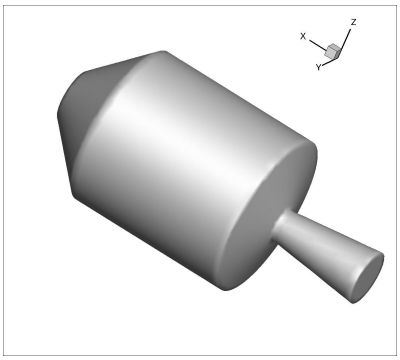


Figure 5.2: Upper Stage mesh

5.2 Input uncertainties

In this study, we consider four primary sources of uncertainties: the material characteristics, the initial flight conditions, the atmosphere model parameters, and the breakup model parameters, for a global number of uncertainties of twenty (20). This list does not include the uncertainty intrinsic to the breakup model. The list of uncertainties is summarized in table 5.2.

5.2.1 Material characteristics

The thermal module uses the fusion temperature (T_{fus}), the fusion enthalpy (H_{fus}), the emissivity (ε) and density (ρ_M). The material characteristics such as the density, the fusion temperature, and the fusion enthalpy are directly measured and therefore are modeled as Gaussian variables and reported in table 5.2. The emissivity uncertainty is more complex to model as it varies along the trajectory. Studies have shown that the reacting flow alters the surface of the aluminum shell during a reentry [6, 59, 8]. While the aluminum emissivity usually varies between 0.3 and 0.5 depending on the surface polishing, due to oxidation, it can reach 0.8. [59]. For this reason, we consider a uniform distribution ranging from 0.3 to 0.8 for the emissivity uncertainty. The influence of oxidation on the other material characteristic uncertainties is neglected. Since the aluminium oxide layer is extremely thin, the mass of oxide is negligible compared to the aluminum mass. Nevertheless, surface changes can significantly affect the object thermal response. The aluminum oxide fusion temperature is much higher than for aluminum. Consequently the radiative heat flux could be significantly underestimated as observed in [8].

5.2.2 Initial conditions

Initially, the upper stage is orbiting on a GTO orbit characterized by its apogee, perigee, inclination, ascending node longitude, perigee argument and true anomaly. The upper stage leaves its initial orbit by rotating and firing a final deorbiting boost using the main thruster.

A deorbiting boost slope at 180 degrees yields a maximum velocity reduction. The deorbiting maneuver conditions are not perfectly known for the following reasons:

- The orbit is not perfectly known, these uncertainties are reflected through independent uniform uncertainties in the orbital elements.
- The deboost amplitude and direction is only partially known (uncertain quantity of propellant, unknown object orientation before deorbiting, etc.); hence they are modeled as uniform distributions.

5.2.3 Atmosphere model parameters

The atmosphere conditions (temperature, density, and pressure) are fluctuating quantities depending on the time of the day, the day of the year, the solar activity or the Earth magnetic activity that are *a priori* unknown in our case. The chosen atmosphere model (MSIS00 [54]) takes as inputs the time of the year in days, the time of the day, the solar activity or the Earth geomagnetic index. We consider all those parameters as random variables. As the time of reentry is unknown, we assume time distributions as uniform distributions over the year. We also model the solar flux intensity and the Earth magnetic activity index as uniformly distributed random variables which ranges are based on historical measurements [207] and reported in table 5.2.

5.2.4 Breakup model parameters

As mentioned in Chapter 3, our probabilistic breakup model uses two uncalibrated (i.e. not estimated using experimental data) parameters T_{frag} and p_{abl}, defining t_{init} and t_{end}. The quantity T_{frag} is the limiting temperature above which the object strength is too weak to withstand aerodynamic forces. Necessarily, T_{frag} is smaller than $T_{fus} = 800K$. Besides, experimental studies found that the strength of aluminum greatly reduces as early as 400K [89, 37]. Many events can trigger fragmentation: the object tumbling can create structural load leading to the buckling of the structure. Local overheating can lead to the thermal dilatation of the material and its buckling. Welding joints may also melt or weaken and fail.

The uniform temperature model used in ADRYANS V4 does not allow to account for the temperature gradient in the upper stage structure that can lead to thermal breakup. It is also not able to detect joints failure that can lead to a breakup. Moreover, the 3 degree of freedom model does not allow to account for the aerodynamic load due to the object attitude motion and the non-uniformity of the heat flux. To account for all the phenomena that cannot be modeled by our physical model, we model T_{frag} as an uncertain parameter. We consider that breakup can occur as soon as the object temperature reaches 400 K depending on the aerodynamic loading or local overheating. Hence T_{frag} is a uniform random variable ranging from 400 K to 700 K.

Similarly, p_{abl} is the minimum percentage of remaining mass such as the object is still intact. As this quantity is not known *a priori* and since it depends on unmodeled breakup phenomena mentioned before, in this work, p_{abl} varies between 50 % and 70 % uniformly.

Although this work considers an unprecedented number of uncertainties, this list is by no means exhaustive. A lot of physics discussed in Chapter 2 is neglected. Some physical aspects such as the erratic attitude motion of the upper stage or complex hypersonic phenomena such as shock interactions, wall chemistry, gas radiation could not be included in the uncertainty analysis although they affect the aerodynamic coefficient and the heat flux. Some model inadequacies, discussed in Chapter 3, are not modeled either: the error induced by the modified Newton method for computing the aerodynamic coefficients, the heat flux correlation formulae, etc. While

Variable	Description	Distribution
ρ_M	material density	$\mathcal{N}(2800., 10.)$ $[kg/m^3]$
ϵ	material emissivity	$\mathcal{U}(0.3, 0.8)$
T_{fus}	fusion temperature	$\mathcal{N}(873., 0.4)$ $[K]$
H_{fus}	fusion enthalpy	$\mathcal{N}(350, 3)$ $[kJ/kg]$
A_{boost}	boost amplitude	$\mathcal{U}(62, 67)$
t_{boost}	boost time	$\mathcal{U}(0, 5)$ $[s]$
α_{boost}	boost inclination	$\mathcal{U}(178, 182)$ $[deg]$
β_{boost}	boost bearing	$\mathcal{U}(-90, 90)$ $[deg]$
apo	initial orbit apogee	$\mathcal{U}(35e6, 35e6 + 35e3)$ $[m]$
per	initial orbit perigee	$\mathcal{U}(254000, 256540)$ $[m]$
inc	initial orbit inclination	$\mathcal{U}(10, 10.1)$ $[deg]$
anl	initial orbit ascending node longitude	$\mathcal{U}(-135, -134.9)$ $[deg]$
pa	initial orbit perigee argument	$\mathcal{U}(130, 130.01)$ $[deg]$
tan	initial orbit true anomaly	$\mathcal{U}(43, 43.01)$ $[deg]$
atmoday	reentry day	$\mathcal{U}[1, 365]$ $[day]$
atmohour	reentry hour	$\mathcal{U}[0, 23]$ $[hour]$
solarf	solar flux	$\mathcal{U}[65, 240]$
ag	magnetic index	$\mathcal{U}[2, 75]$
T_{frag}	fragmentation temperature	$\mathcal{U}[400, 700]$ $[K]$
p_{abl}	ablation percentage	$\mathcal{U}[0.5, 0.7]$

Table 5.2: List of uncertainties

the model inadequacy indeed induces an error in the QoI, they are also extremely challenging to model. Firstly, because they are usually unknown and have to be estimated quantitatively with high fidelity models or experimental data. Secondly, because they typically have to be represented with high dimensional models and therefore they could not be included in this study already featuring twenty uncertainties.

5.3 Uncertainty propagation strategy

In this section, we construct a surrogate model to approximate the mapping between the uncertainties ξ and the quantities of interest such as the flight conditions at breakup, the object thermal characteristics at several altitudes or the flight conditions at t_{init} and t_{end}. The surrogate model will then be used to propagate the uncertainties in the system described in Fig. 5.1. Out of the four solvers represented in Fig. 5.1 only two are approximated with a SoGP: the deorbiting solver and the trajectory/thermal solver. The aerodynamic solver is solely used to construct the aerodynamic coefficient table and does not depend on the input uncertainties ξ. Therefore it is not included in the framework. The breakup solver is fully analytical and hence cheap to evaluate. For this reason, including it into the SoGP presents no advantage.

Depending on the QoI we consider, the breakup solver is not always necessary. In section 5.4.4, for instance, we compute the convective heat flux at several altitudes. Hence, we do not use the breakup solver. In this case, we can directly apply the SoGP framework to the deorbiting solver composed with the trajectory solver.

On the other hand, when making the breakup predictions with the breakup solver, the construction of a surrogate model is more complex. In this case, we need to build the mapping

between the input uncertainties ξ and the object trajectory that is then used by the breakup solver to derive the breakup conditions. For a given uncertainty value ξ_i, the resulting trajectory is denoted $(X(t_j, \xi_i))_{j\in[0,m]}$ for the position and $(V(t_j, \xi_i))_{j\in[0,m]}$ for the velocity. The dimension of $(X(t_j, \xi_i))_{j\in[0,m]}$ maybe large, depending on the time sampling rate.

If we were to apply directly the SoGP framework to this problem, we would have to build a SoGP with the order of m outputs. The computational burden of building this surrogate model would be prohibitive. To overcome this difficulty, the trajectories are represented on a reduced basis obtained with the Karhunen-Loéve Expansion presented in 4.1.4. We interpret the $(X(t_j, \xi_i))_{i,j}$ as the realizations of a stochastic process. The decomposition expresses the large matrix $(X(t_j, \xi_i))_{i,j}$ in a reduced basis such that :

$$(X(t_j, \xi_i))_{i,j} \simeq \mu(t_j) + \sum_{k=1}^{N} \sqrt{\lambda_k}\zeta_k(\xi_i)\phi_k(t_j) \tag{5.1}$$

where the ϕ_k are the eigenfunctions that depend on time only and $(\zeta_k)_k$ the coefficients that only depend on the uncertain inputs ξ. The $(\lambda_k)_k$ are the eigenvalues. Similar expressions are obtained with $(V(t_j, \xi_i))_{j\in[0,m]}$.

The breakup solver also requires the temperature evolution $T(t)$ and the mass evolution $m(t)$ to compute t_{init} and t_{end}. For the temperature evolution $T(t)$ and the mass evolution $m(t)$, the KLE did not yield a good approximation due to their non stationary behaviors. The object mass for instance is constant over time until its temperature reaches fusion temperature and then its mass starts decreasing. This type of behaviors is not well capture by the KLE. To bypass this problem, we modify the inputs of the breakup solver. Instead of providing the entire trajectory, temperature and mass evolution, we directly provide the object trajectory between t_{init} and t_{end} to the breakup solver. It then samples uniformly along this fragment of trajectory to derive the breakup conditions and does not need the mass and temperature evolution.

Since t_{init} and t_{end} depend on ξ, all the fragment trajectories do not have the same duration. Hence we proceed to a time scaling of each trajectory. As a consequence, with this time scaling, the absolute breakup time information is lost but it is still possible to sample uniformly on this modified time scale and recover the breakup distribution.

Finally, the surrogate model used to estimate the object trajectory is the composition of a SoGP and a KLE. The construction of this surrogate model is detailed in algorithm 6 while the evaluation step is given in 7. Note that the algorithms are written for the position, but the same procedure applies for the velocity vector.

The slight drawback of this approach is in line 2 of algorithm 6 : by performing the time rescaling, we loose the absolute time information. This is obvious in the evaluation step as algorithm 7 returns $X(\tilde{t})_{\tilde{t}\in\hat{\mathcal{T}}}$ instead of $X(t)_{t\in\mathcal{T}}$. Nevertheless, this information loss has no consequence on the breakup solver that can still sample at breakup time uniformly in $\hat{\mathcal{T}}$ instead of $\mathcal{T}$ and get the same breakup flight condition distribution. Moreover, if needed, the time scaling could also be learnt by the surrogate model

The advantage of this procedure is that we do not have to reconstruct the mass and temperature evolutions that could not be easily approximated with a truncated KLE. The accuracy of the surrogate model is quantified in section 5.4.1.

Algorithm 6 Surrogate model construction procedure

procedure ConstructSurrogate($(\xi_i)_{i\in[1,n]}, (X(t_j,\xi_i))_{i\leq n, j\leq m}$)
  for $i = 1,\dots,n$ **do**
    Truncate the trajectory between t_{init} and t_{end} : $(X(t,\xi_i))_{t\in\mathcal{T}}$, where $\mathcal{T} = [t_{init}(\xi_i), t_{init}(\xi_i)+\Delta t,\dots,t_{end}(\xi_i)]$
    Construct the time rescaled trajectory $\left(\hat{X}(\tilde{t})\right)_{\tilde{t}\in\hat{\mathcal{T}}}$ with $\tilde{t} = \frac{t-t_{init}}{t_{end}-t_{init}}$ and $\hat{\mathcal{T}} = [0,\Delta t,\dots,1]$such that $\hat{X}(\tilde{t}) = X(t)$
    Rescale the trajectory with $\tilde{X}(\tilde{t},\xi_i) = \frac{\hat{X}(\tilde{t},\xi_i)-X_{init}(\xi_i)}{X_{end}(\xi_i)-X_{init}(\xi_i)}$
  Construct a KLE of the $\left(\tilde{X}_i(\tilde{t}_m)\right)_{i,m}$ using 4.1.4 and return the mean function μ, eigenfunctions $(\phi_k)_{k\leq N}$ and coefficients $(\zeta_k(\xi_i))_{k\leq N, i\leq n}$
  Train a SoGP to learn the mapping between the ξ_i and the KLE coefficients $(\zeta_k(\xi_i))_{k,i}$ and the flight conditions at t_{init} and t_{end} : $X_{init}(\xi_i)$,$X_{end}(\xi_i)$
  return SoGP,$(\phi_k)_{k\leq N}$,μ

Algorithm 7 Surrogate model evaluation procedure

procedure EvalSurrogate(ξ, SoGP, $(\phi_k)_{k\leq N}$, μ)
  Evaluate the flight conditions at $X_{init}(\xi)$,$X_{end}(\xi)$ and the KLE coefficients $(\zeta_k(\xi))$ using the SoGP
  Reconstruct the reduced trajectory $\tilde{X}(\tilde{t}) = \mu(\tilde{t}) + \sum_{k=1}^{N}\sqrt{\lambda_k}\zeta_k(\xi)\phi_k(\tilde{t})$
  Recover the unscaled trajectory $X(\tilde{t}) = X(\tilde{t})\times(X(t_{end}) - X(t_{init})) + X(t_{init})$
  return $X(\tilde{t})_{\tilde{t}\in\hat{\mathcal{T}}}$

5.4 Results

In this section, we present the results of the uncertainty propagation obtained with the surrogate model proposed in section 5.3 after having quantified the errors in the surrogate model prediction.

5.4.1 Surrogate model convergence

An accurate surrogate model should be constructed and validated before being used to obtain the distribution of the QoI. In this section, we perform an analysis of the convergence of the SoGP strategy illustrated in this work. In particular, we compare the performance of the SoGP and a single classical GP used as surrogate models. We illustrate the convergence of the upper stage altitude computed at t_{init}. The L2 error is calculated using 500 validation samples and normalized by the output standard deviation. Fig. 5.3 shows the evolution of the L2 error of the two surrogates models trained with the same LHS training plans.

The results show that the SoGP framework outperforms the single GP by almost an order of magnitude. Using 1000 training samples, the SoGP reaches an error of 0.1 % of the output standard deviation. We observe a similar behavior for the predictions of other quantities of interest presented in this work. This low level of error justifies the use of the SoGP as a surrogate model of the system. Monte Carlo sampling technique can then be used with the SoGP to compute the QoI statistics and distribution and carry out sensitivity analysis, at a negligible computational cost.

To highlight the gain of building a surrogate model compared to standard Monte Carlo approaches, Table 5.3 summarizes the results obtained when computing the mean and the variance of the object altitude at t_{init}, using a fixed computational budget of 600 solver evaluations

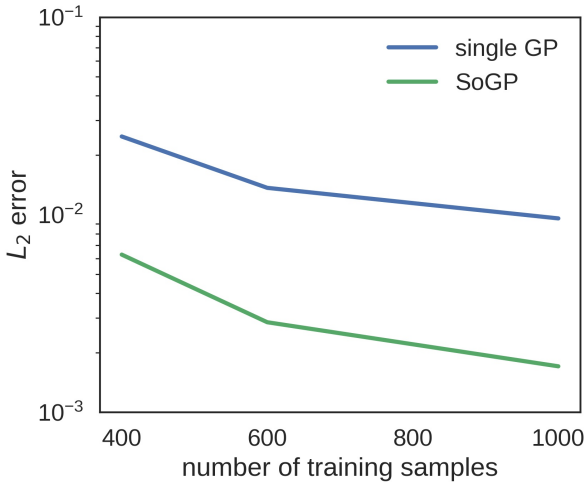


Figure 5.3: L2 error evolution for a SoGP and single GP (BB) surrogate model

Method	altitude mean $[m]$	error interval $[m]$	altitude std $[m]$	error interval $[m]$
Standard MC	71407	(300,299)	4455.	(-207, 217)
Single GP	71400	(-10, +10)	4335	(-7,+7)
SoGP	71412	(-10, +10)	4328	(-7, +7)

Table 5.3: Estimation of the first two statistical moment altitude distribution at t_{init} using 600 SoS evaluations using Monte Carlo, a single GP trained on 600 samples using evaluated 500 000 times or a SoGP trained on 600 samples using evaluated 500 000 times.

representing a cost of around one hour of computation on 28 cores IntelXeon at 2.4 GHz. We compare three different methods. i) The standard Monte Carlo approach that directly uses the 600 samples to estimate the mean and variance. ii) The single GP and iii) the SoGP use the 600 samples to build a surrogate model and then run the surrogate model with 500 000 samples at no additional computational cost. The 500 000 samples evaluated using the surrogate models are then used to compute the mean and variance of the altitude. Additionally, a Bayesian statistical error estimate [208] for each estimation is included in Table 5.3. Note that the surrogate model-based approaches are orders of magnitudes more precise than the standard Monte Carlo approach since many more samples (500 000 *vs.* 600) could be used with the surrogate assisted strategies for the same computational budget.

When comparing two surrogate model-based approaches, the best indicator remains the L2 error computed on a validation set (Fig. 5.3). In the rest of this section, the results presented are obtained using a SoGP trained with 600 evaluations.

For the breakup predictions, we assess the accuracy the surrogate model composed of the truncated KLE and the SoGP In addition to the SoGP accuracy, we also assess the error induced by the KLE over the trajectory computed by the trajectory and thermal solve The overall absolute errors estimated on a validation set of 500 additional trajectories are reported in table 5.4 and are much smaller than the observed uncertainties. The error includes the contribution from the expansion truncation, the SoGP error in the KLE coefficients and the approximate estimation of the eigenvalues and eigenfunctions.

Variable	absolute error
Altitude[m]	20
longitude [deg]	0.02
latitude [deg]	0.002
Velocity [m/s]	0.9

Table 5.4: L2 absolute error averaged over the whole trajectory

5.4.2 Uncertain breakup range

As stated in Section 3.2.2 (see Eq. 3.5), t_{init} and t_{end} are crucial times in the breakup prediction.

t_{init} represents the earliest moment at which breakup can occur and defines the lower bound of the time interval where breakup occurs. Physically, at t_{init}, the upper stage reaches a critical temperature for which the upper stage shell is weakened and may not resist to aerodynamic forces.

t_{end} is the other bound delimiting the time interval. At t_{end} a significant portion of the upper stage envelope has been ablated, and the chances that the upper stage is still intact are meagre. As a consequence, we assume that breakup has occurred by that time.

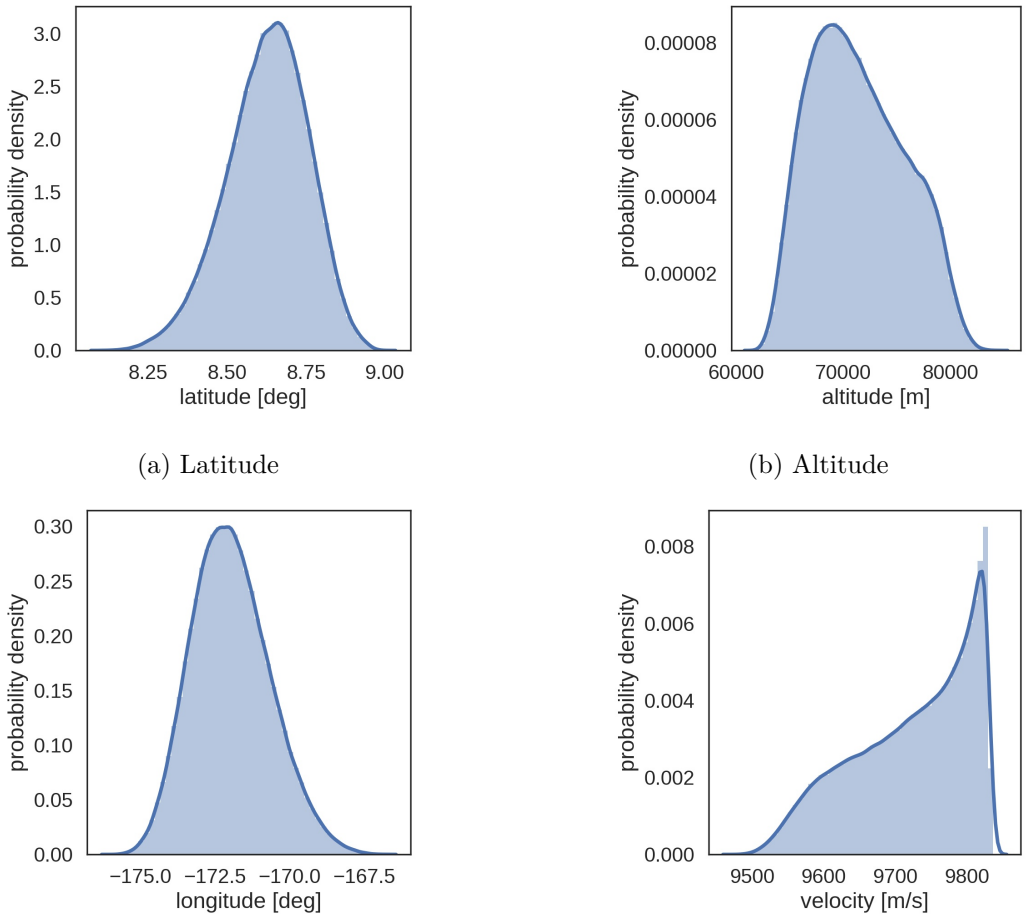
5.4.2.1 t_{init} prediction

In this section, the objective is to compute the uncertainties in altitude, longitude, latitude and velocity at t_{init}. In Table 5.5, the principal statistics of the distributions of altitude, longitude, latitude and velocity at t_{init} are reported, namely, the mean, the standard deviation and different quantiles at 0.5 % 2.5% 50 % 97.5 % and 99.5%. Quantiles are cut points in a distribution delimiting intervals of same probability. They provide additional information about the shape of the distribution compared to the mean and standard deviation, in particular, low and high quantiles provide information about the tails of the distribution. The distribution of altitudes is widespread since 99 % of the distribution (between the 0.5 % and 99.5 %) lies between 63km and 81km with a mean and a median around 71km of altitude. The results show that breakup cannot occur before 81km.

Note that studies considering the reentry of upper stage Delta II or Automated Transfer Vehicle (ATV) use a breakup altitude at 78 km [75, 49] but most object-oriented software use a reentry around 75-85 km [38]. Owing to the nature of GTO orbit, the slope at which the upper stage reenters is high compared to typical reentry of LEO trajectories presented in [75, 49]. As a consequence, the breakup altitude can be lower than the standard 78 km value usually selected for a LEO satellite reentry. Fig. 5.4 represents the probability density function of the latitude (5.4a), of the altitude (5.4b), of the longitude (5.4c) and of the velocity (5.4d) at t_{init}.

	Altitude [m]	Longitude [deg]	Latitude [deg]	Velocity [m/s]
mean	71412	-171.97	8.627	9722.8
std	4327	1.3	0.129	81.9
0.5 %	63352	-174.84	8.26	9524.
2.5 %	64367	-174.29	8.35	9553.
50 %	70962	-172.05	8.64	9739
97.5 %	79804	-169.22	8.86	9828
99.5 %	81223	-168.37	8.91	9831

Table 5.5: Main statistics of the flight conditions at t_{init}



(a) Latitude

(b) Altitude

(c) Longitude

(d) Velocity

Figure 5.4: Distributions at t_{init}

In Fig. 5.4b, the altitude distribution is skewed toward low altitudes. Due to the exponential increase in the atmosphere density as altitude decreases, the received heat flux given and the object temperature increase more rapidly at low altitudes than high altitudes. The convective heat flux scales as the velocity to the cube such that the object velocity is a direct factor leading to heat production and temperature increase. Hence high-velocity cases lead to high heat flux and high altitudes at t_{init}. Consequently, there is a positive correlation between the velocity and the altitude of t_{init}. The velocity distribution is also highly skewed toward high speeds that correspond to cases where t_{init} is reached at high altitudes. Longitude and latitude distributions are relatively symmetrical although slightly skewed toward high latitudes and low longitudes.

Fig. 5.5 represents the first order Sobol indices for altitude, longitude, latitude and velocity. For all four variables, the same inputs account for most of the variance while the majority of the inputs have no significant influence on the output variance. The predominant variables are T_{frag} which is the temperature at t_{init} and the deorbiting maneuver uncertainties such as the boost orientation and amplitude. The variance in altitude and velocity are mostly due to T_{frag} (respectively 89 % and 95%). The rest is due to the boost amplitude and orientation uncertainties. For latitude and longitude, T_{frag} explains only 10 % of the output variance while

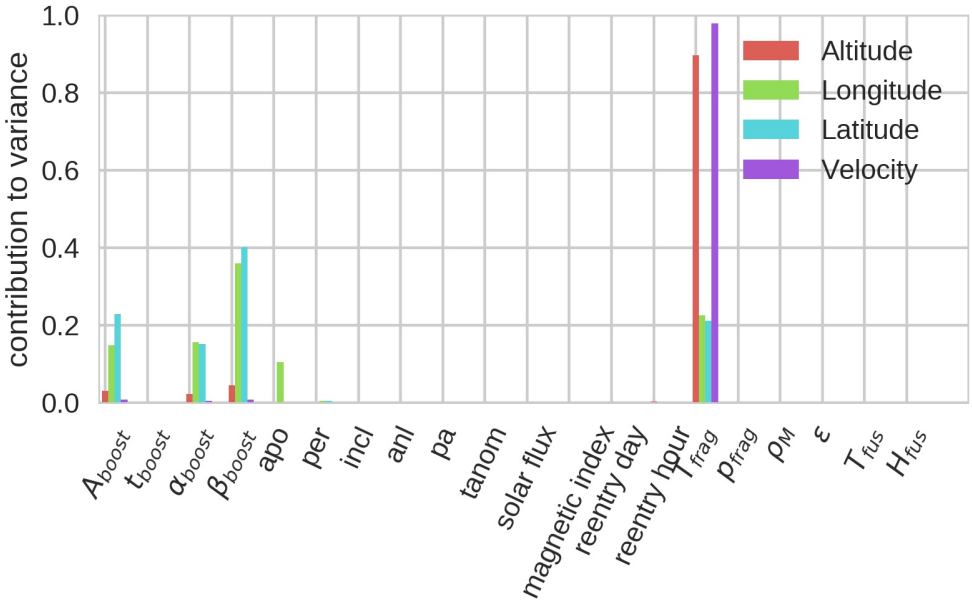


Figure 5.5: Sensitivity indices at t_{init}

most of the variance is due to the uncertainties associated with the deorbiting boost. Note that the uncertainty in the apogee position has a significant influence in the longitude variance only. Although relatively small, the reentry time affects the altitude at t_{init}. The atmosphere density is dependant on the reentry time in particular at high altitude, and in turn, it changes the convective heat flux.

5.4.2.2 t_{end} prediction

The time t_{end} is the latest time at which the object can breakup, therefore the object is very likely to have fragmented before reaching t_{end}. As for t_{init}, table 5.6 summarizes the distribution characteristics of altitude, longitude, latitude and velocity at t_{end}. The altitude distribution ranges from 51780 to 65750 meters such that the support of the distribution is a bit smaller than for t_{init}. As for the t_{init} case, the altitude pdf represented in Fig. 5.6b is slightly skewed toward low altitudes. Concerning the velocity, the uncertainties have significantly increased since the standard deviation is 81 m/s at t_{init} and 173 m/s at t_{end}. The shape of the distribution plotted in Fig. 5.6d is also very different for the two times. At t_{end}, the PDF is much less skewed but still positively correlated to the altitude. The uncertainties in longitude and latitude are also larger at t_{end} than at t_{init} as shown in Fig. 5.6a and 5.6c. Contrary to t_{init}, at t_{end}, the longitude and latitude are highly correlated with the velocity and the altitude. High velocity cases are likely to lead to high altitude breakup than slower one. Finally note that the object velocity remains extremely high even at altitudes as low as 50 km.

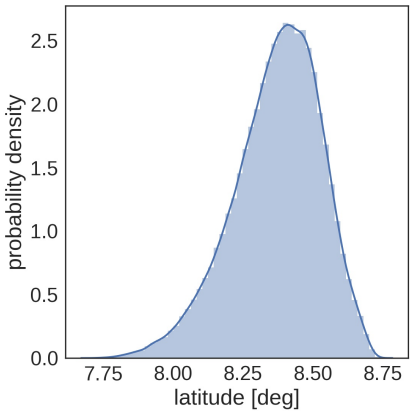
Figure 5.7 shows the first order Sobol sensitivity indices for the four quantities of interest at t_{end}, namely altitude, latitude, longitude and velocity. The deorbiting maneuvering uncertainties are the predominant sources of uncertainties for all quantities of interest. In the case of velocity (and altitude to a lesser extent), the uncertainty in the breakup model parameter p_{frag} explains 15 % and 2 % of the output variances. As for t_{init}, the uncertainties in the apogee position of the initial orbit influences the longitude at t_{end}. While the range of variation of p_{frag} is voluntarily

	Altitude [m]	Longitude [deg]	Latitude [deg]	Velocity [m/s]
mean	56714.	-169.35	8.37	8644.
std	2778.	1.45	0.16	173
0.5%	51782.	-172.2	7.9	8247
2.5%	52466.	-171.8	8.0	8323
50%	56289.	-169.5	8.4	8637
97.5%	63260.	-166.2	8.6	9001
99.5%	65750.	-165.1	8.7	9097

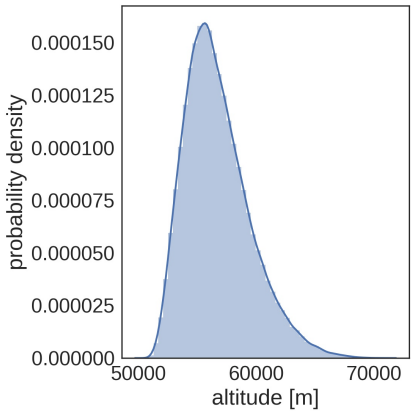
Table 5.6: Main statistics of the flight conditions at t_{end}

chosen very large (between 50 % and 70% of the shell mass) for robustness, its influence is still secondary compared to the deorbiting uncertainties. This result contrasts with the sensitivity results at t_{init} where most of the uncertainty comes from the uncalibrated T_{frag} parameter model. Note that the uncertainties in the material characteristics have a minor influence in the QoIs, even at t_{end} and despite the vast uncertainty range used to model the emissivity changes due to surface oxidation. This result shows that the material characteristics are known well enough compared to the other sources of uncertainties.

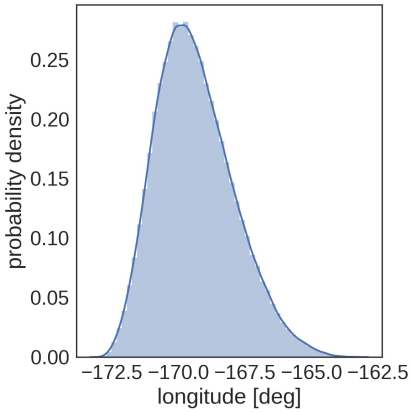
As expected, the altitude range breakup covered by our breakup model is widespread from altitudes as high as 81km down to 52km. The sensitivity analysis showed that most of the uncertainty in the estimation of the flight conditions for t_{init} and t_{end} is mostly due to the poorly calibrated breakup model parameters p_{abl} and T_{frag}, and the deorbiting manoeuvre uncertainties. To improve the predictive ability of the model, the model parameters should be calibrated either with experimental results or high fidelity simulations of the breakup.



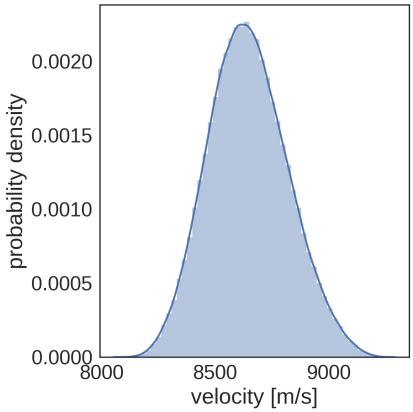
(a) Latitude



(b) Altitude



(c) Longitude



(d) Velocity

Figure 5.6: Flight conditions distributions at t_{end}

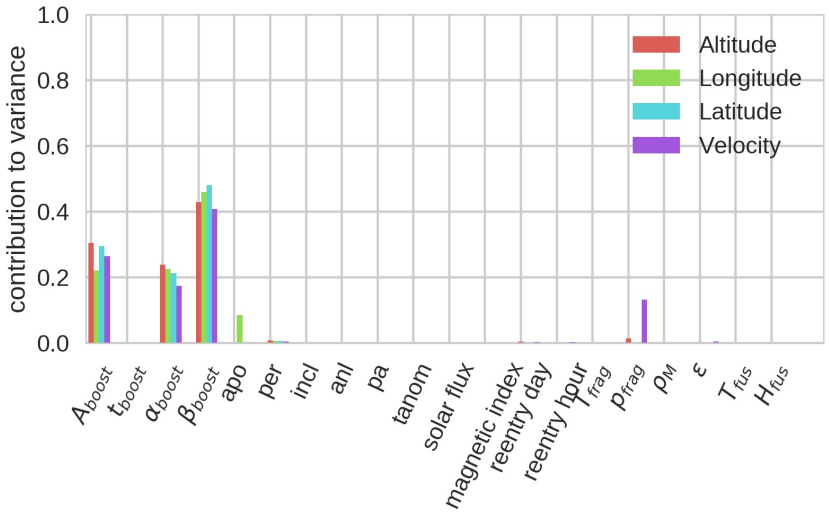


Figure 5.7: Sensitivity indices at t_{end}

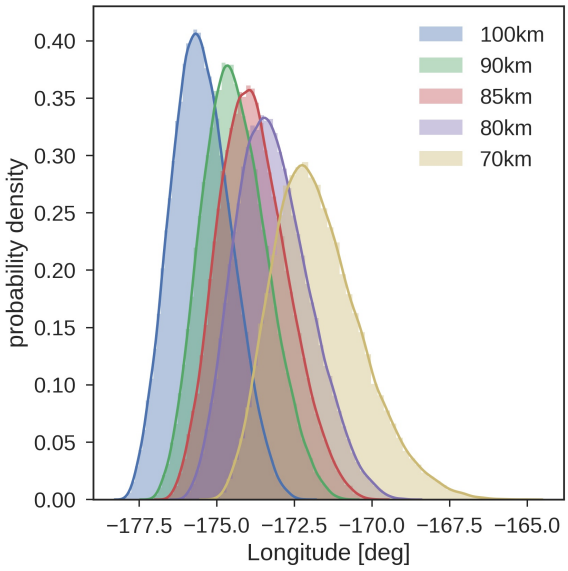


Figure 5.8: Longitude distribution at 100km, 90km, 85km, 80km, 70km

5.4.3 Uncertainty evolution along the trajectory

In this section, we consider the evolution of the uncertainties in the position, velocity and thermal properties of the upper stage along the trajectory where the breakup has a low probability of occurring (i.e. before 70km). The objective is to understand better how the uncertainty in the object position evolves with time. We consider several altitudes: 100km, 90km, 85km, 80km and 70km. In each case, we observe the longitude, latitude, velocity, convective heat flux and radiative heat flux.

Longitude In Fig. 5.8, the pdf for longitude at the altitudes of 100km, 90km, 85km, 80km and 70 km are represented. While at 100km, the pdf is fairly symmetrical, it becomes larger and more skewed as the object falls on Earth. The sensitivity analysis presented in Fig. 5.9 shows that the longitude exclusively depends on the deorbiting uncertainties (mostly the boost orientation and the boost amplitude). While it is not surprising that the material characteristics do not influence the position of the object before the breakup, it is interesting to observe that the atmosphere uncertainties also have no influence on the longitude at any altitude. The sensitivity indices vary with altitude significantly. The uncertainties in the initial orbit (apogee and perigee positions) have a significant influence at high altitudes, but their influence reduces at lower altitudes. Meanwhile, the impact of the deorbiting boost amplitude increases as the altitude decreases.

Latitude Fig. 5.10 illustrates the pdfs of the object latitude for different altitudes. As for the longitude, the distributions become wider and more skewed as the altitude decreases. Fig. 5.11 illustrates the sensitivity indices for the different altitudes. As for the longitude, most of the uncertainty comes from the unknown deorbiting conditions. On the other hand, the initial orbit uncertainties have a much smaller influence on the latitude than the longitude. Note that further analysis showed that the first order Sobol indices decrease with the altitude, while higher order interactions between the deorbiting boost orientation angles increase.

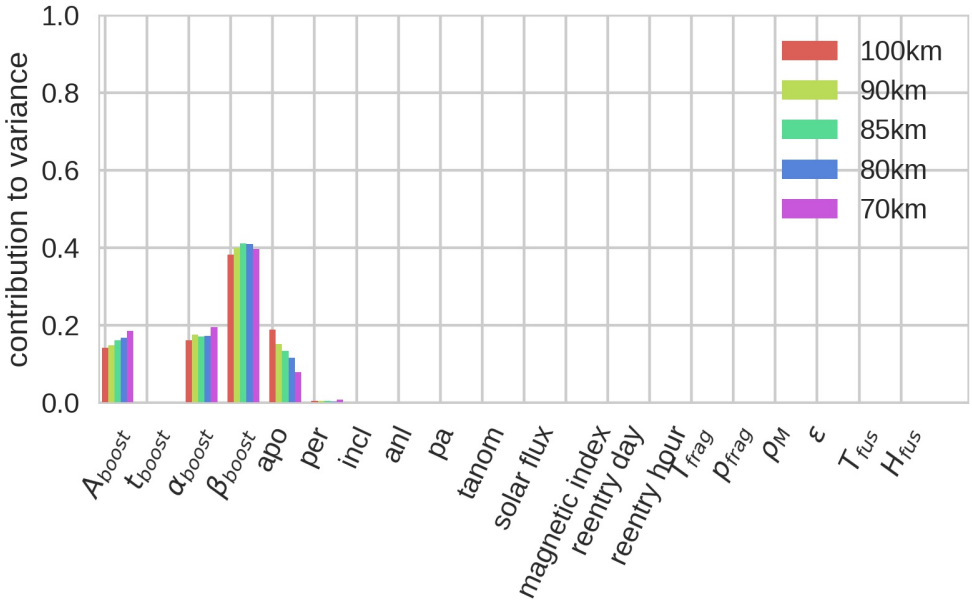


Figure 5.9: Longitude sensitivity index at 100km, 90km, 85km, 80km, 70km

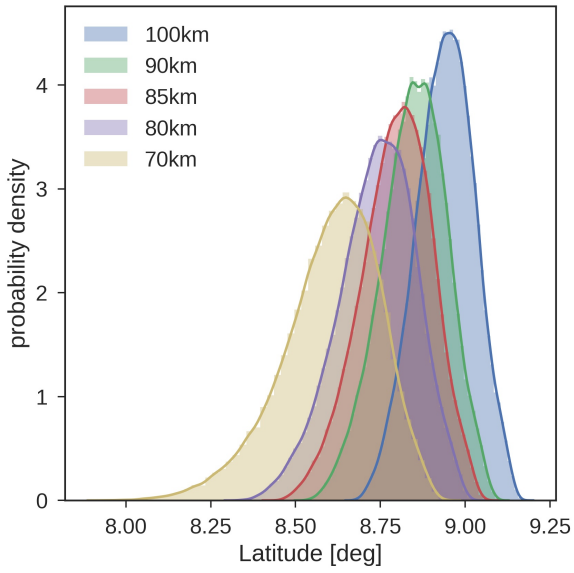


Figure 5.10: Latitude distribution at 100km, 90km, 85km, 80km, 70km

Velocity The reentry considered here is a high-speed reentry compared other human-made object reentries. For instance reentries from LEO would be around 6.5 km/s to 7km/s while this one is around 9.5 km/s. Fig. 5.12 represents the distributions of the velocity at different altitudes. At 100km, the uncertainties in the velocity are very small as the distribution features a standard deviation of 1.72 m/s and a mean value of 9833 m/s. As the object falls, the standard deviation slightly increases from 1.8 m/s at 90 km to 2.2 m/s at 85km and then 3.9 m/s at 80 km. Note that the object mean velocity also increases between 100 km and 85 km from 9833 m/s to 9840 m/s. For this range of altitude, the atmospheric density is extremely low, and the

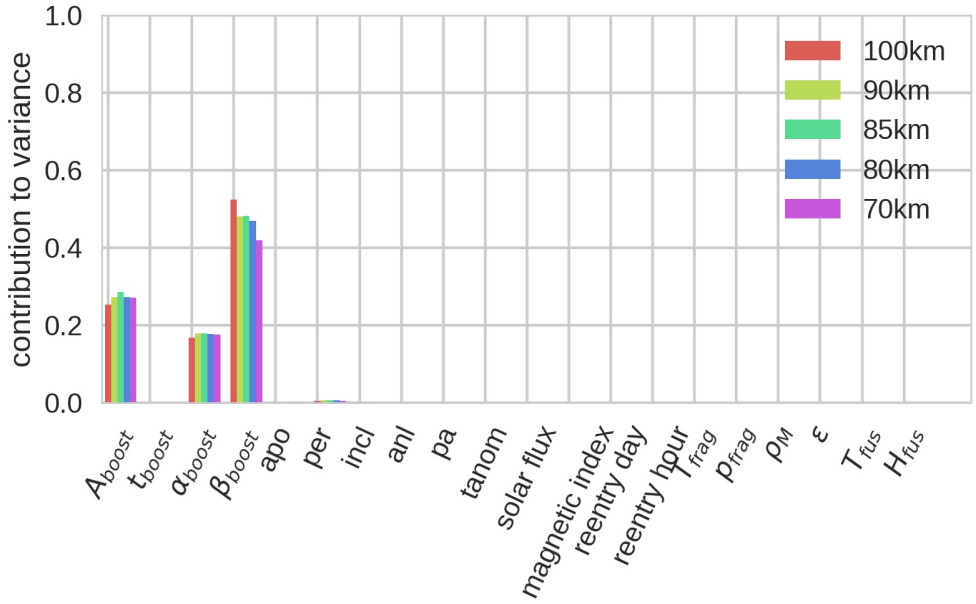


Figure 5.11: Latitude sensitivity index at 100km, 90km, 85km, 80km, 70km

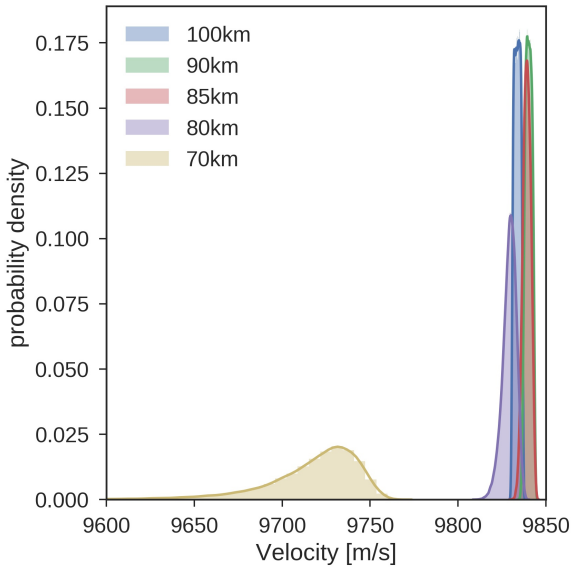


Figure 5.12: Velocity distribution at 100km, 90km, 85km, 80km, 70km

Earth gravity force is larger than the drag, resulting in an increase in velocity until an altitude around 80 km where the object meets denser layers of the atmosphere and starts slowing down due to atmospheric drag. At 70km, the object velocity reduces to 9717 m/s on average, but the uncertainty greatly increases as shown in Fig. 5.12. We observe a highly skewed distribution with a heavy tail toward low values around 9400 m/s.

The sensitivity indices presented in Fig. 5.13 are very different, depending on the altitude considered. At 100km, the uncertainties in the apogee position of the initial orbit are predominant

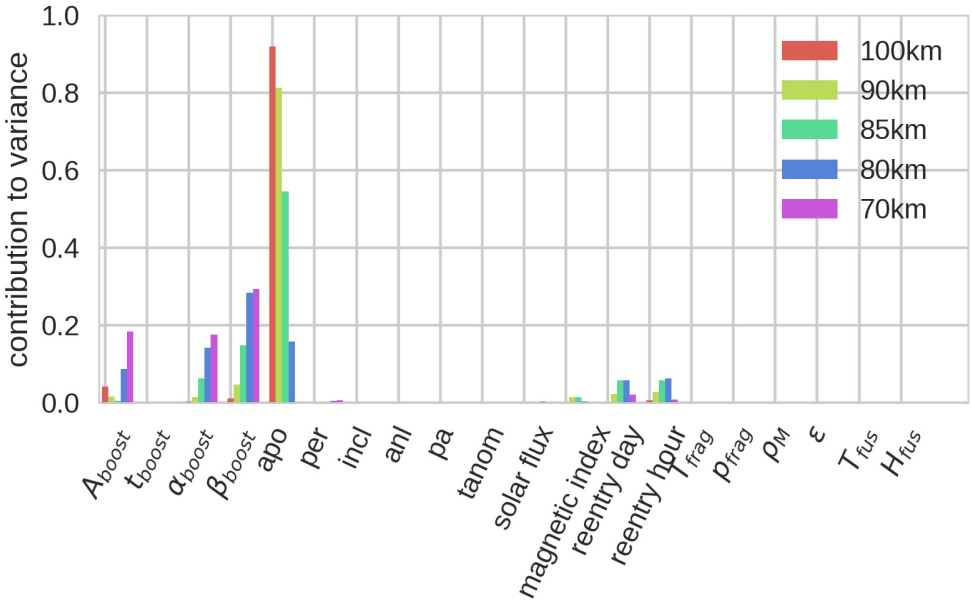


Figure 5.13: Velocity sensitivity index at 100km, 90km, 85km, 80km, 70km

with over 90 % of the altitude variance explained, but it plummets as the object reaches lower altitudes to become negligible at 70km eventually. The uncertainties in the deorbiting maneuver also gain in influence as the object falls back on Earth. One interesting exception is the deorbiting boost amplitude that is nearly 0 at 85km. At this altitude, the object reaches its maximal velocity due to the gravity acceleration and the thin atmosphere. Later, when the object meets the denser layers of the atmosphere, the atmospheric uncertainties have more and more influence on the velocity due to atmospheric drag. This effect is directly reflected in the sensitivity indices of the time of reentry and to a lesser extent in the geomagnetic index and solar flux. Note that the atmospheric density and temperature are particularly uncertain for high altitudes above 80 km but the become significantly smaller at lower altitudes [3]. For this reason, the influence of the sources of uncertainties related to the atmosphere starts decreasing when reaching 70 km. Note that at 70 km the first order Sobol indices only represent 70 % of the total variance. The residual variance comes from higher-order interactions between the boost amplitude and the boost orientation.

5.4.4 Uncertain thermal quantities

The uncertainties in the object temperature have a significant impact on our ability to accurately predict breakup since the breakup model presented is exclusively based upon thermal considerations. In this section, we focus on three significant quantities computed at different altitudes of the trajectory of the upper stage, namely, the convective heat flux, the radiated heat flux and the object temperature.

From 120 km to 70 km, the convective heat flux keeps increasing as the upper stage meets denser and denser layers of the atmosphere. At 70 km it reaches 42 MW. Fig. 5.14 reports the pdf of the convective heat flux distributions on a log scale. Although the distribution supports seem to shrink as the altitude decreases due to the log scale, the standard deviation doubles from a 364 kW at 100km to 670 kW at 70 km.

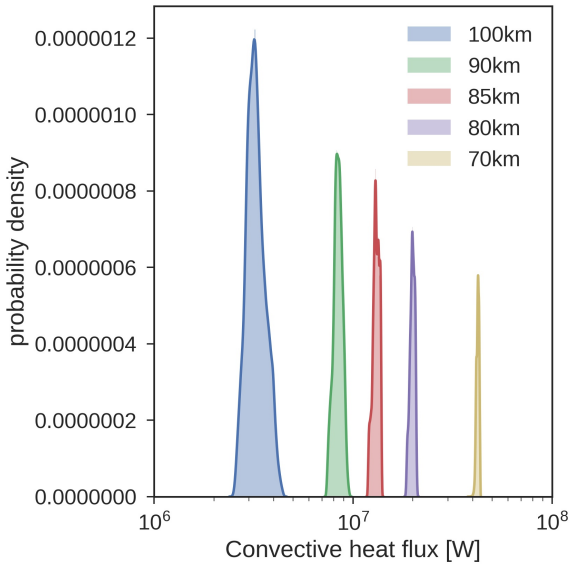


Figure 5.14: Convective heatflux distributions at 100km, 90km, 85km, 80km, 70km

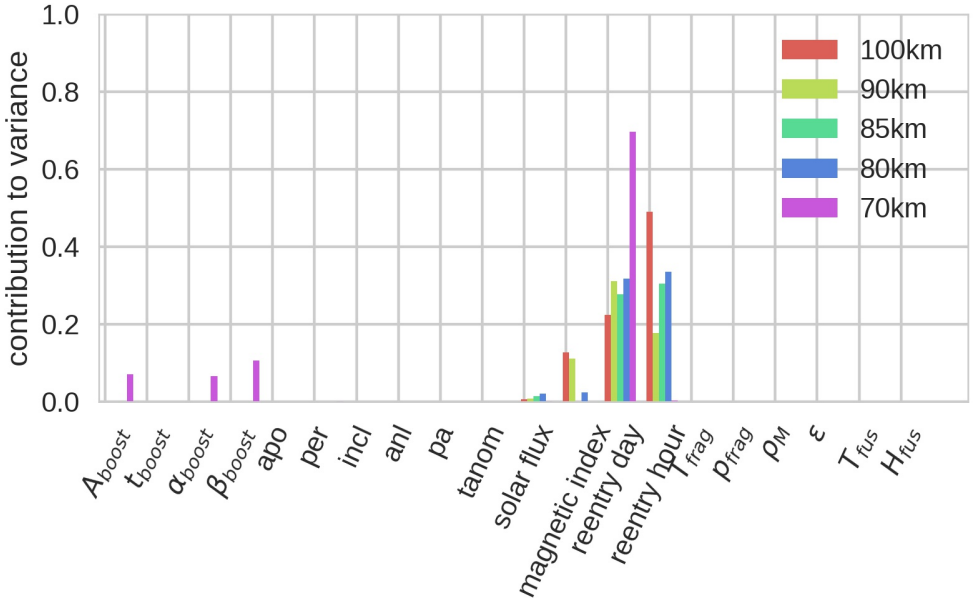


Figure 5.15: Convective heatflux sensitivity indices at 100km, 90km, 85km, 80km, 70km

The Sobol indices plotted in Fig. 5.15 show that most of the convective heat flux variance comes from uncertain atmosphere model parameters, namely the solar flux, the Earth magnetic index and the time of reentry. When the object is above 80km, the time of reentry accounts for around 70-80 % variance while the rest is due to the Earth magnetic activity. At 70 km, the uncertainties associated with the atmosphere, except the reentry day, lessen. Simultaneously, the uncertainties in the deorbiting maneuver conditions become more and more dominant influencing, in particular, the velocity of the object and therefore the convective heat flux.

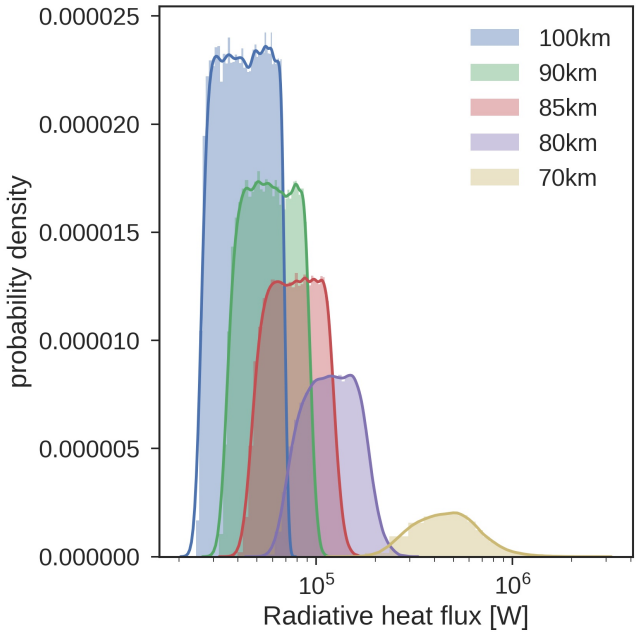


Figure 5.16: Radiated heatflux distributions at 100km, 90km, 85km, 80km, 70km

The second heat flux contribution of ADRYANS V4 (see eq. A.7), is the radiated heat flux. The heat radiated by the object is directly dependent on the temperature and the emissivity of the object through Stefan Boltzmann law (Eq. 2.5). We report the pdf of the heat flux radiated at different altitudes on a log scale in Fig. 5.16. As for the convective heat flux, it increases as the altitude decreases. Note that it remains one order of magnitude smaller than the convective heat flux at this state of the reentry. At 100km, the radiative heat flux distribution is almost uniform, but it shifts toward a more bell-shaped curve as the altitude decreases.

The sensitivity analysis results summarized in Fig. 5.17 show that most of the variance is due to the emissivity variance. This influence is strong at 100km, and it diminishes with the altitude to reach less than 40 % of the variance at 70 km. This observation is coherent with the uniform-like shape of the distribution above 80 km since the emissivity, modeled as a uniform distribution, is directly proportional to the heat flux (Eq. 2.5). The influence of the deorbiting maneuver uncertainties become more and more important as the altitude decreases to reach a total of 45 % at 70 km. The atmosphere uncertainties have limited influence in the radiated heat flux around 90 km and they become completely negligible at 70km. At 80 km the first order Sobol indices represent 80 % of the total variance. The rest is due to higher-order interactions between the deorbiting boost orientation uncertainties α_{boost} and β_{boost}.

Figure 5.18 shows the evolution of the object temperature distribution at different altitudes. The uncertainties increase as the object reaches lower altitudes. In fact, at 100 km of altitude, 95 % of the probability density lies between 307 and 315 K whereas at 80 km it ranges between 518 K and 677 K. Note that at this altitude, the distribution is skewed toward high temperatures.

The sensitivity analysis presented in Fig. 5.19 also changes with altitude. For high altitudes, the time of reentry is the largest contributor to variance with more than 85 % of the variance explained at 100km but its influence vanishes to only a few percents of the variance at 70 km. On

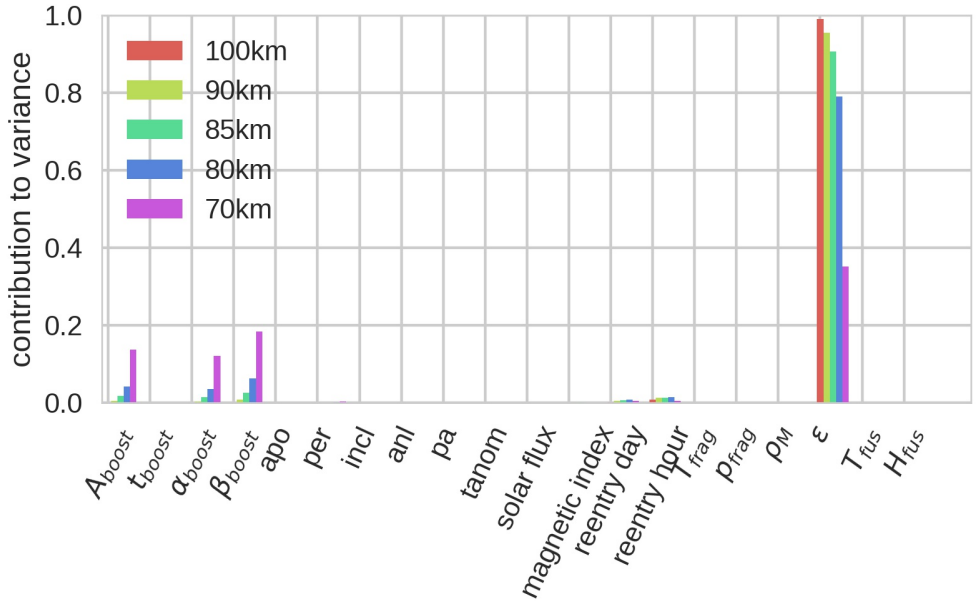


Figure 5.17: Radiated heatflux sensitivity index at 100km, 90km, 85km, 80km, 70km

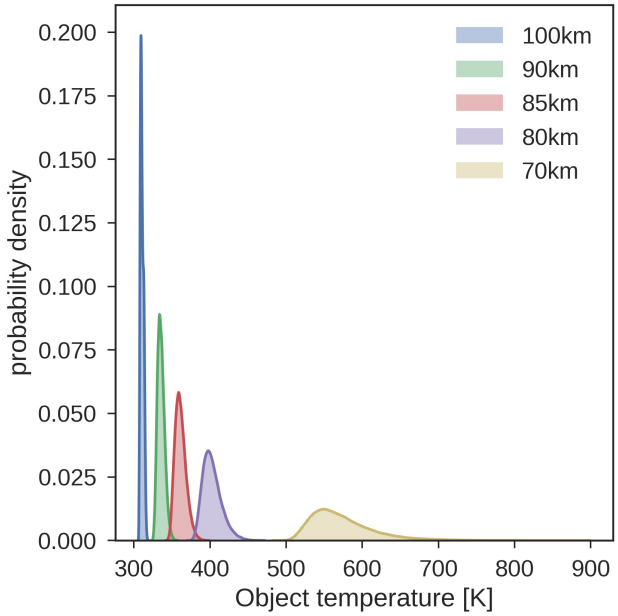


Figure 5.18: Object temperature distributions at 100km, 90km, 85km, 80km, 70km

the other hand, the uncertainties in the deorbiting boost become more and more important as the object moves along the trajectory. Note the minimal influence of the initial orbit uncertainties and in particular the perigee position. The uncertainty in the object emissivity does not influence temperature for this portion of the trajectory although it dramatically affects the radiated heat flux. This observation comes from the fact that the temperature increase is mostly due to the convective heat flux that is one order of magnitude higher than the radiated heat flux.

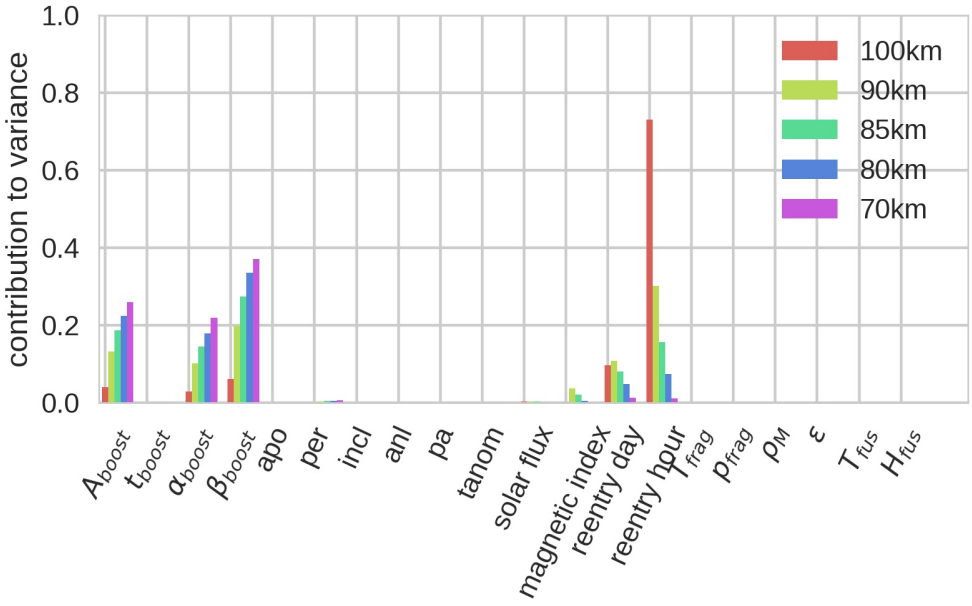


Figure 5.19: Object temperature sensitivity indices at 100km, 90km, 85km, 80km, 70km

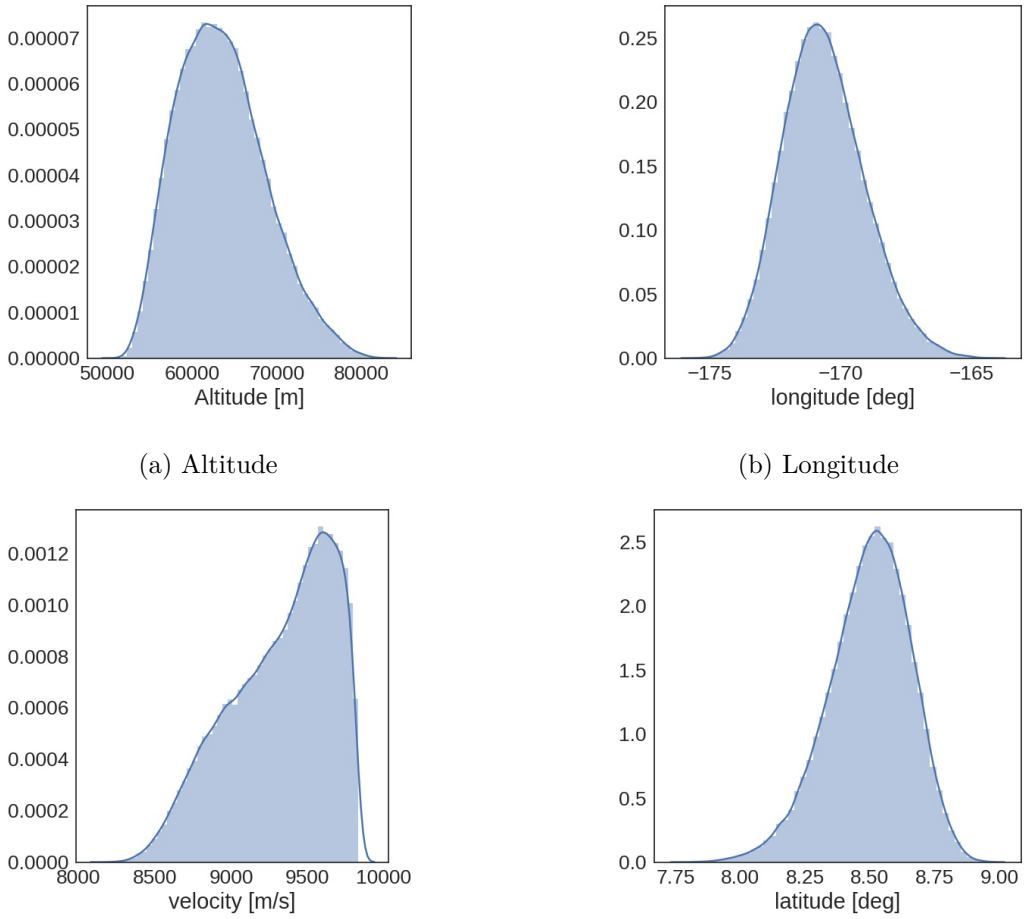
5.4.5 Uncertain breakup conditions

The objectives of this work include deriving the flight conditions when breakup occurs. Those quantities are of primary interest for survivability tools in order to compute the on-ground risk. In this work, the breakup model bypasses complex and expensive computational models and assumes it occurs at a random time t_{frag} between t_{init} and t_{end} according to a uniform distribution. There are therefore two sources of randomness in this model. The first one comes from the input uncertainties ξ that characterize the distribution of $t_{init}(\xi)$ and $t_{end}(\xi)$ presented in Section 5.4.2.1 and 5.4.2.2. The second level of randomness comes into play when t_{frag} is sampled from $\mathcal{U}(t_{init}(\xi), t_{end}(\xi))$ (see eq. 3.5). In other words

$$t_{frag} = t_{init}(\xi) + u_{frag}(t_{end}(\xi) - t_{init}(\xi)) \tag{5.2}$$

where u_{frag} is a uniform random variable between 0 and 1 that represents the second level of uncertainty. To compare the influence of the probabilistic aspect of the model with the other uncertainties, the random variable u_{frag} is viewed as a random input of the breakup model and is included in the sensitivity analysis.

At breakup, the quantities of interest are the altitude, longitude, latitude and velocity (their distributions are represented in Figs. 5.20a 5.20b, 5.20d and 5.20c). According to our model, the breakup occurs at random between t_{init} and t_{end} according to a uniform distribution. The range of possible breakup goes from 50 km to more than 80 km. The most probable breakup altitude is around 62km. Certain studies assume a constant breakup at 80-78 km of altitude [49]. While this study confirms that, in the case of a GTO re-entry of a upper stage, a breakup can take place at 80km, it is more likely to happen at lower altitudes. Note that the breakup velocity is extremely high ranging from 8250 m/s to 9800 m/s, in comparison with the average reentry velocity at 120 km of 9800 m/s. High-velocity breakups usually occur at high altitude for reentries with relatively low slope while the low-velocity scenarii lead to breakups at lower altitudes.



(a) Altitude

(b) Longitude

(c) Velocity

(d) Latitude

Figure 5.20: Flight conditions distributions at breakup

Figure 5.21 summarizes the results of the sensitivity analysis. The main contributors to the breakup conditions can be divided into two groups: the deorbiting conditions (A_{boost},α_{boost},β_{boost}) and the breakup model parameters (T_{frag}, p_{frag} and u_{frag}). Recall that u_{frag} represents the uncertainty due to the random nature of the breakup model (see eq. 5.2). For the altitude and velocity, the main contributors are u_{frag} and to a lesser extent T_{frag} which triggers t_{init}. Hence, the stochastic nature of the breakup model is the most significant contribution to the velocity and altitude variance compared to the other uncertain inputs. Altogether, u_{frag} and t_{init} explain almost 80 % of the variance in the altitude and more than 90 % of the velocity variance. For those two variables, it is clear that the breakup model is responsible for most of the uncertainties and that higher fidelity breakup models could reduce the uncertainties significantly. Regarding the longitude and the latitude of the breakup, the analysis of variance shows that the breakup model parameters account for 30% of variance while the rest comes from the deorbiting boost orientation and amplitude. For longitude, a few percents of the variance come from uncertainties in the apogee altitude of the initial orbit.

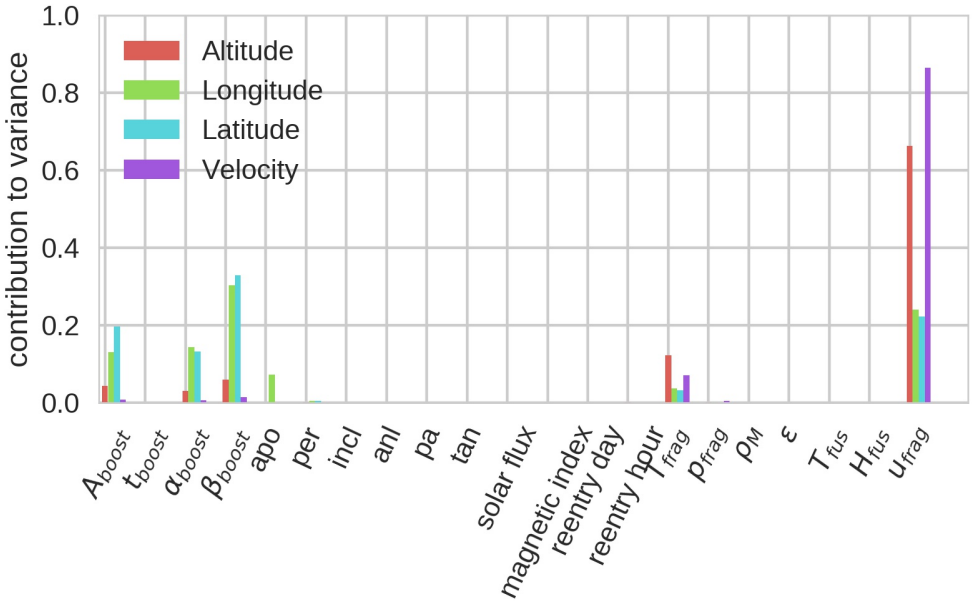


Figure 5.21: Sensitivity indices at breakup

5.5 Conclusion

In this work, we tackle the problem of predicting space object breakup. A probabilistic approach is chosen to model the uncertainties. In particular, we have shaped two levels of uncertainty. In the first one, a stochastic breakup model is proposed and implemented to predict the reentry of an Upper Stage deorbited from a GTO orbit. The second level deals with a large number of uncertain input model parameters. We supplement the simulator with uncertainty quantification tools from chapter 4 that efficiently compute probability distributions of the breakup conditions and other critical quantities at different reentry times at a reasonable computational cost. The sensitivity analysis completes the uncertainty analysis with the identification of the most critical uncertain variables. By simultaneously considering a large number of uncertainties, we can compare the influence of each uncertainty on the output.

We have investigated the evolution of the uncertainties along the upper stage trajectory and the breakup condition distributions. For the object longitude, the predominant uncertainties are the initial orbit characteristics whereas the velocity, the longitude and the latitude are mostly influenced by the deorbiting boost maneuver uncertainties. The material characteristic uncertainties have a minor influence in this scenario showing that the material models are relatively precise compared to other sources of uncertainties, despite the large uncertainty range selected for the aluminium emissivity. This result is explained by the low contribution of radiative heat flux to the total heat flux. Different results would be expected with materials with higher fusion temperature. Regarding the breakup prediction, the analysis shows that for reentries from a GTO orbit, the breakup occurs between 80 and 52 km. The variance-based sensitivity analysis showed that the breakup model parameter uncertainties are responsible for most of the uncertainties when computing the object flight conditions at breakup. This result suggests that the breakup conditions could be more precisely known by calibrating the model on experiments or refining the solver with high fidelity models.

The predictions could be improved by considering the high fidelity model presented in the appendix B. The high fidelity model features a six degree of freedom trajectory solver to account

for the object attitude motion and a more realistic model for the object thermal response. In the next chapters, we focus on assessing the survivability of the fragments released after the breakup using the survivability solver presented in Chapter 3

Chapter 6

Surrogate Modelling for Risk Estimation

After the object breakup, the fragments are released. In some cases, the fragments may be disintegrated entirely in the atmosphere depending on the material composition and the amount of energy they receive from the external flow. In other cases, the fragments survive and impact the Earth surface. The survivability solver assesses the object survivability and computes the impact location if any. Based on the survivability solver output, several risk metrics can be derived: the impact zones, the casualty area or the expected number of victims.
In this chapter, we propose a novel surrogate modeling approach to emulate the survivability solver and predict the expected number of victims. The mapping between the number victims and the input uncertainties is viewed as the composition of a classifier that assesses the survivability of a fragment and a continuous function that maps the input uncertainties to the impact locations if the object survives. This formulation permits to construct a surrogate model of the survivability solver using a classifier and a GP. Additionally, an active learning method aiming at reducing the estimated risk error is presented and tested on analytical tests and a toy reentry problem. The contributions of this chapter are :

- *A novel formulation of the survivability problem and the expected number of victims (section 6.1);*
- *A dedicated active learning strategy to reduce the estimated risk error (section 6.3);*
- *A series of analytical tests and a fragment reentry prediction test to validate the method 7.5.*

6.1 Problem formulation

The space debris mitigation guidelines suggest several measures to protect human assets despite the uncertainties associated with the prediction. The expected number of victims is a risk metric that can be estimated from population density maps and the object impact location. The guidelines define the on-ground risk *as the expected number of victims, or the probability of having at least one victim, with a threshold value typically set to* 10^{-5} [30]. In this chapter, we focus on the computation of an accurate estimation of the risk distribution considering the system uncertainties.

We are interested in both outcomes of a survivability solver: state if the object survives upon reentry and if it is the case, the impact location on the Earth surface. As in the previous

chapters, ξ is the vector of random inputs, with values in $\Omega \subset \mathbb{R}^n$ and probability density function p_ξ. For a given input condition ξ, the solver returns the information disintegrated or not disintegrated and in the latter case the impact location. In this work, we model the prediction of the survivability solver as the output of two functions B and Y. We have $B : \Omega \mapsto \{0, 1\}$ and $Y : \Omega_b \subseteq \Omega \mapsto \mathcal{Y}$ where $\Omega_b = \{\xi \in \Omega | B(\xi) = 1\}$. The function B assesses the survivability of the space object for a given ξ: if the object burns it returns 0 otherwise it returns 1. Y is a mapping between the input ξ and the impact location if the object survived. The graphical representation of the model is given in Fig. 6.1. Hence the definition of Y depends on B. The output space $\mathcal{Y}$ is the impact location space expressed in terms of longitudes and latitudes. Hence we have $\mathcal{Y} \subseteq [-180, 180] \times [-90, 90]$.

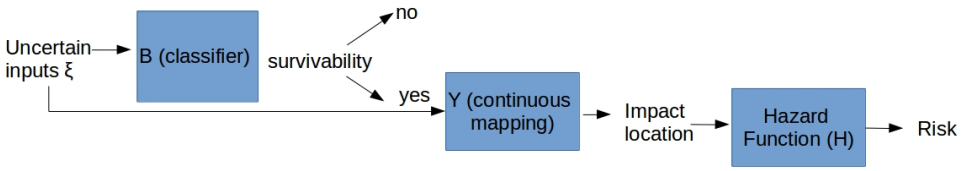


Figure 6.1: Graphical representation of the risk estimation.

To compute the risk, we define the hazard function $H : \mathcal{Y} \mapsto \mathbb{R}^+$. The hazard function depends on the fragment casualty area and the human population density. The hazard function is assumed to be finite on $\mathcal{Y}$ and continuously derivable. For a particular fragment, H depends on the impact location only. Hence the risk for a given fragment is defined as $R : \Omega \mapsto \mathbb{R}^+$:

$$R(\xi) = \begin{cases} 0 & \text{if } B(\xi) = 0 \text{ (the object burns up)} \\ H(Y(\xi)) & \text{if } B(\xi) = 1 \text{ (the object reaches the ground)} \end{cases} . \tag{6.1}$$

Recall that Y is not defined for $\xi \in \Omega$ such that $B(\xi) = 0$.

Ultimately, we are interested in computing the statistics of $R(\xi)$ such as its expectation, $\mathbb{E}_\xi [R(\xi)]$, variance, $\mathbb{V}_\xi [R(\xi)]$, or its quantiles. Monte Carlo methods can estimate these statistics, but since each sample requires a solver evaluation, the computational burden may become formidable, especially for low quantiles. A solution is to construct a surrogate model of R with a limited number of solver evaluations. The surrogate can then be used in place of R to generate Monte Carlo samples. In [49] for instance, they used a polynomial approximation of the survivability solver to estimate the distribution of the impact location, while the survivability distribution was not derived. Similarly, [52] proposes to use kriging surrogate models to estimate the impact zone assuming that the object always survives upon reentry. However, to the best of our knowledge, the construction of surrogate models that emulate the survivability solver for both the survivability and the impact location remains an open problem. A priori, the risk function R is discontinuous, and the polynomial or standard kriging based methods are ill-suited.

In this work, we propose an original approach that constructs a surrogate model suitable to predict both the survivability and the impact location, and that can adequately address the discontinuity in the risk response. The construction relies on the use of Gaussian Processes for regression and classification. Our approach builds a surrogate model that accurately approximates the expected number of victims as defined in Chapter 2. High levels of accuracy are achieved by using an active learning method specifically designed to reduce the error in the risk estimation.

In Section 6.2, we introduce our surrogate model construction strategy, in Section 6.3 we present our active learning strategy. Finally, in Section 6.4, we test the surrogate model construction strategy on an analytical problem and a simple object reentry problem.

6.2 Surrogate model construction

In this section, we illustrate how to construct $\tilde{R}$, which is an approximation of R. We construct two surrogate models of Y and B denoted as $\tilde{Y}$ and $\tilde{B}$. One critical point when building $\tilde{Y}$ is the input space characterization. The function Y is defined on Ω_b that depends on B. Since B is an unknown function we wish to approximate, Ω_b is also unknown. Instead, we propose to build a surrogate model of an arbitrary extension of Y, $\hat{Y} : \Omega \mapsto \mathcal{Y}$ such that $\hat{Y}(\xi) = Y(\xi) \ \ \forall \xi \in \Omega_b$. We further define

$$\hat{R}(\xi) = H(\hat{Y}(\xi))B(\xi). \tag{6.2}$$

Note that $\hat{R}(\xi) = R(\xi) \ \ \forall \xi \in \Omega$ regardless of the values taken by $\hat{Y}$ outside of Ω_b. The value of $\hat{Y}$ outside of Ω_b can be set to an arbitrary finite constant as it is multiplied by 0 outside Ω_b.

We propose to build $\tilde{R}$ as an approximation of $\hat{R}$. To do so, we construct a GP $\tilde{Y}$ and a GP classifier $\tilde{B}$ (see section 4.1.4) surrogate models of $\hat{Y}$ and B such as $\tilde{R}$ is defined as :

$$\tilde{R}(\xi) = H(\tilde{Y}(\xi))\tilde{B}(\xi) \tag{6.3}$$

This formulation presents several advantages. First, it is cheap to evaluate since the hazard function H is algebraic and $\tilde{Y}(\xi)$,$\tilde{B}(\xi)$ are GPs, therefore $\tilde{R}$ is also inexpensive. Second, compared to a surrogate model that would directly try to learn R, the proposed formulation bypasses the problem of building a surrogate model of a discontinuous function.

Note that one could construct a surrogate model of $H \circ \hat{Y}$ instead of just $\hat{Y}$. The resulting approximation of the risk would not be more advantageous than ours in terms of evaluation computational cost since the hazard function is purely algebraic. On the other hand, we believe that our approximation would feature lower level of approximation error. The function H is challenging to approximate because population density is extremely unevenly distributed. There are empty regions (for instance oceans) where the population density is null and highly populated regions (cities) where H is large. The function H is expected to have localized high-intensity peaks and nearly flat responses otherwise. This behaviour makes the surrogate $H \circ Y$ unlikely to perform well compared to a surrogate model of $\hat{Y}$. This is why we prefer to build a surrogate model of $\hat{Y}$ and directly use H.

We use Gaussian Processes (GP) for regression and classification to construct $\tilde{Y}$ and $\tilde{B}$ (see section 4.1.4). We define $\mathcal{X}_B$ and $\mathcal{X}_Y$ as two subsets of Ω, containing the input training samples for $\tilde{B}$ and $\tilde{Y}$. Note that $\mathcal{X}_Y \subset \mathcal{X}_B$. Using the $\tilde{Y}$ and $\tilde{B}$ constructed, we obtain:

$$\tilde{R}(\xi) = H(\tilde{Y}(\xi))\tilde{B}(\xi). \tag{6.4}$$

As $\tilde{Y}$ and $\tilde{B}$ are stochastic, so is $\tilde{R}$. To make predictions, we define

$$\tilde{R}_{\text{pred}}(\xi) = H(\mu_Y(\xi)) \times 1_{\bar{\pi}(\xi)>0.5}. \tag{6.5}$$

The variance of $\tilde{R}(\xi)$ is given by :

$$\begin{aligned}
\mathbb{V}\left[\tilde{R}(\xi)\right] &= \mathbb{V}\left[H(\tilde{Y}(\xi))\right]\mathbb{E}\left[\tilde{B}(\xi)^2\right] + \mathbb{E}\left[H(\tilde{f}(\xi))\right]^2\mathbb{V}\left[\tilde{B}(\xi)\right] \\
&= \mathbb{V}\left[H(\tilde{Y}(\xi))\right]p(\xi) + \mathbb{E}\left[H(\tilde{Y}(\xi))\right]^2(1-p(\xi))p(\xi) \qquad (6.6)\\
&\simeq \nabla H(\mu_Y(\xi))\Sigma_Y(\xi)\nabla H(\mu_Y(\xi))^T p(\xi) + H(\mu_Y(\xi))^2(1-p(\xi))p(\xi), \qquad (6.7)
\end{aligned}$$

The new variable p is the membership probability. It is estimated in two different ways that will be explicated later. The first term in Eq. (6.6), denoted $V_Y = \mathbb{V}\left[H(\tilde{Y}(\xi))\right] p(\xi)$, corresponds to the variance induced by $\tilde{Y}$ used to predict the object impact location whereas the second term, denoted $V_B = \mathbb{E}\left[H(\tilde{Y}(\xi))\right]^2 (1 - p(\xi))p(\xi)$ corresponds to the contribution to variance of the classifier $\tilde{B}$. If $\mathbb{V}\left[\tilde{R}(\xi)\right]$ is interpreted as the uncertainty in the predicted risk, each of the terms in eqs. 6.6 and 6.7 corresponds to the contributions from $\tilde{Y}$ and $\tilde{B}$.

As mentioned earlier, p can be evaluated in two different ways, depending on the sigmoid function that is used in the GP classifier (see section 4.1.4). In this work, we consider the probit function and the Heaviside function $I(x) = 1_{x>0}$. We define:

$$p_1(\xi) = \int \frac{1}{1 + \exp(l(\xi))} p(l)dl \tag{6.8}$$

and

$$p_2(\xi) = \int 1_{l(\xi)>0} p(l)dl = \Phi\left(\frac{\mu_l(\xi)}{\sigma_l(\xi)}\right) \tag{6.9}$$

where Φ is the CDF of a standard Gaussian and l is the latent GP defined in section 4.1.4. With p_2, the predictions are harder than with p_1. The two probabilities are implemented in Eq. (6.7) and the resulting predicted variance are denoted

$$V_1 = \nabla H(\mu_Y(\xi))\Sigma_Y(\xi)\nabla H(\mu_Y(\xi))^T p_1(\xi) + H(\mu_Y(\xi))^2(1 - p_1(\xi))p_1(\xi) \tag{6.10}$$

and

$$V_2 = \nabla H(\mu_Y(\xi))\Sigma_Y(\xi)\nabla H(\mu_Y(\xi))^T p_2(\xi) + H(\mu_Y(\xi))^2(1 - p_2(\xi))p_2(\xi). \tag{6.11}$$

Note that the latent GP posterior is estimated using expectation propagation and a probit function in both cases (see section 4.1.4 for details.

6.3 Risk-based active learning strategy

In this work, we propose an active learning criterion to improve both the classifier $\tilde{B}$ and $\tilde{Y}$ to get a precise estimate of the ground risk R. We suppose to have already initial training sets $\mathcal{X}_Y$ and $\mathcal{X}_B$ and we wish to enrich with additional training samples. Using $\mathcal{X}_Y$ and $\mathcal{X}_B$, $\tilde{R}(\xi)$ is constructed. The objective is to minimize the integrated L2 error. To estimate the L2 error, we use the predictive variance derived in Eq. (6.7). Hence the averaged L2 error is:

$$\epsilon^2 = \int_\Omega \mathbb{V}\left[\tilde{R}(\xi)\right] p_\xi(\xi)d\xi. \tag{6.12}$$

The objective is to add training points to reduce the quantity ϵ^2. We define the training distribution G valued in Ω with probability density distribution f_G. We further define:

$$f_G(\xi) = Z_G \mathbb{V}\left[\tilde{R}(\xi)\right] p_\xi(\xi), \tag{6.13}$$

where Z_G is a normalizing constant such that f_G integrates to one over Ω. Additional training samples are generated by sampling from G. Once samples have been added to $\mathcal{X}_B$ and $\mathcal{X}_Y$, $\tilde{R}$ should be updated. The number of samples to be added to $\mathcal{X}_B$ and $\mathcal{X}_Y$ between two model updates is problem dependent. The most accurate approach would be to update the model $\tilde{R}$ and f_G after every sample addition. However, in certain cases, the solver can be evaluated

in parallel on k processes so that the computational cost of adding one or k points is similar to adding 1. In this case, it may be preferable to sample k points from f_G between two updates.

Sampling from the distribution with non-normalized pdf is a well-studied problem. Two classical ideas were tested in this work: the rejection sampling and the MCMC sampling [209]. The rejection sampling was eventually selected for its simplicity and robustness to multimodal distributions.

In section 4.1.5, we observed that most AL strategies require the resolution of an optimization problem (IMSPE, MMSPE, etc.). Our approach however is inherently stochastic and does not require to solve an optimization problem. It is also well suited for classification problems. One alternative we tested for this problem was to sequentially add training points where the predicted variance was maximum. The issue with this approach is that the predictive variance is always maximal along the classifier boundary, where the risk is maximum. As a result, the active learner systematically added samples at the same location. This approach is not only inefficient for reducing the error but it also creates numerical instabilities when training the GPs. By sampling training points from G, we aim at finding a balance between sampling in very high uncertainty regions that represent a small volume of Ω and less uncertain regions that represent a larger volume in Ω. In our approach, we hope to sample in high uncertainty regions as long as the probability weight in these regions is significant. Otherwise, the active learner will more likely improve the prediction in less uncertain but more probable (or larger) regions.

6.4 Results

In this section, we assess the performance of the proposed approach for constructing an accurate surrogate model of R on an analytical case and an atmospheric reentry test case. In the analytical case, we consider three hazard functions featuring a uniform, unimodal and multimodal behavior.

In the uniform case, the hazard function is constant over $\mathcal{Y}$ such that the second term in (6.6) is zero. As a consequence, the accuracy of the impact location predictor $\tilde{Y}$ does not influence the accuracy in $\tilde{R}$. The problem reduces to a classification problem where the impact location does not matter; only the survivability information affects the risk. The two other hazard functions (unimodal and multi-modal hazard) vary in space. For those cases, the second terms in Eq. (6.6) (or (6.7)) may dominate in regions where the hazard function varies rapidly.

For each hazard function, the following four training approaches are compared:

- The *naive* approach generates training samples using the distribution of ξ regardless of $\tilde{Y}$ and $\tilde{B}$.
- The *AL-Risk* approach is a risk-based learning strategy where the active learner enriches the initial training sets with samples with high predicted risk. More specifically the additional training points are sampled from the random variable D which pdf is equal to:

$$f_D(\xi) = Z_D \tilde{R}_{\text{pred}}(\xi) p_\xi(\xi) \tag{6.14}$$

  where Z_D is a normalizing constant. This active learning strategy is similar to the strategy presented in section 6.3. The difference is that the *AL-Risk* approach adds points where the estimated risk is high and in section 6.3, samples are added where the predicted risk variance is high.

- The *AL-Probit* strategy is the active learning method presented in section 6.3 with the variance estimate $\mathbb{V}\left[\tilde{R}(\xi)\right] = V_1$ (see Eq. (6.10)).

- The *AL-Heaviside* strategy is the active learning method presented in section 6.3 with the variance estimate $\mathbb{V}\left[\tilde{R}(\xi)\right] = V_2$ (see Eq. (6.11)).

For the approaches proposed in this work (AL-Probit, AL-Heaviside), we test the influence of adding several training samples at once to take advantage of the potential parallel evaluation of a solver. The AL strategies proposed here depend on the current training set through the predictive variance. However, assuming that the distribution remains almost unchanged with the addition of a few training points, several training points could be added between two updates of the predictive variance and the sampling distribution H. The advantage of using such an approach is the possibility to run the expensive solver in parallel. To verify the influence of adding several samples between two updates, we considered three cases. In the first case, the predictive variance is updated at each additional training sample. In the second and third cases, the predictive variance updated every 10 and 50 additional training samples, respectively. If not stated otherwise, the numerical experiments are repeated ten times for the analytical 2D case and the mean performance is reported.

The second test-case presented here deals with the risk associated with the reentry of a debris over the Pacific ocean. The hazard function used in this case is multimodal, it corresponds to a case where the object would fall over highly populated islands made up for this study.

In all cases, the error in the estimated risk is defined as :

$$Err = \sqrt{\int_{\Omega} \left(\tilde{R}(\xi_i) - R(\xi_i)\right)^2 dp_{\xi}(\xi) d\xi}. \tag{6.15}$$

It is then approximated with

$$Err = \sqrt{\frac{1}{N_v} \sum_{i}^{N_v} \left(\tilde{R}(\xi_i) - R(\xi_i)\right)^2} \tag{6.16}$$

where $(\xi_i)_{i \leq N_v}$ is the validation set sampled from ξ. For the analytical test $N_v = 5,000,000$ while for the application $N_v = 100,000$

6.4.1 2D test case with several risk functions

In this test case, we consider two uncertain parameters, independent, uniformly distributed between 0 and 1. In particular, we have $\Omega = [0,1]^2$. Survivability mapping is defined as:

$$B : (\xi_1, \xi_2) \longrightarrow \begin{cases} 1 & \text{if } f(\xi_1, \xi_2) = \xi_2 - 0.05 - 0.3\sin(0.1\xi_1) - 0.3\cos(1.1\xi_1)\exp(-1.3\xi_1^2) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{6.17}$$

The impact location mapping Y, defined on $\Omega_b = \{(\xi_1, \xi_2) \in \Omega \mid b(\xi_1, \xi_2) = 1\}$ is defined as :

$$Y : (\xi_1, \xi_2) \longrightarrow \left(\sin(4\xi_1^2)\xi_2^2 + \exp(-\xi_1)\sin(\xi_1)\exp(-\xi_1(1+\xi_1)), \xi_1^2 + \frac{1}{(1+\exp(3\xi_1\xi_2))}\right), \tag{6.18}$$

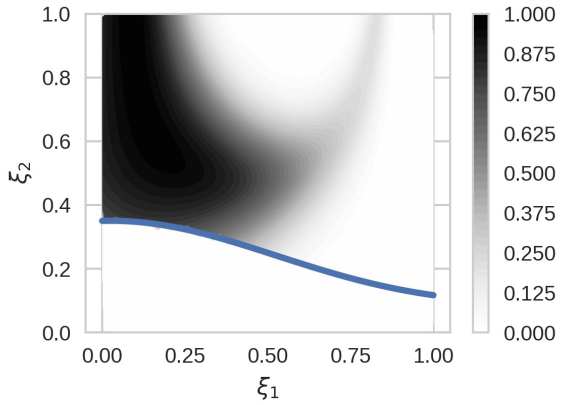


Figure 6.2: Representation of R and the survivability boundary associated to B (blue line) and corresponding to $f(\xi_1, \xi_2) = 0$

Unimodal hazard function

The first test is performed with the unimodal hazard function defined as :

$$H : (y_1, y_2) \in F \longrightarrow \exp\left(-15((y_1 - 0.3)^2 - (y_1 - 0.3)(y_2 - 0.4) + (y_2 - 0.4)^2)\right). \tag{6.19}$$

Using the definition of the risk function from Eq. 6.1, the risk is plotted in Fig. 6.2. For clarity, the survivability boundary is represented. As expected, the risk is null on one side of the boundary and varies on the other side depending on the risk value.

The evolution of the L2 error between the risk R and the estimated risk $\tilde{R}$ is presented in Fig 6.3. The curves represent the results obtained with the naive, AL-Risk, AL-Probit and AL-Heaviside approaches. The curves are the average prediction over ten realizations of the experiment except for the naive approach for which 25 realizations were considered. The results clearly illustrate the advantage of using one of the active learning strategy developed in this work compared to a purely uniform strategy (denoted *naive*) or one based on the risk intensity. To highlight the difference in performance, the AL-Risk and naive are tested with larger training sets than active learning strategies (up to 1000 samples *versus* 350 samples for the AL-probit and AL-Heaviside).

The AL-probit and AL-Heaviside strategies bring a significant error reduction with a much higher convergence rate than the naive and AL-Risk approaches. From figure 6.3,we observe that the AL-Heaviside is a bit more accurate than the AL-probit but the difference is limited. In this test, the error in the risk comes from misclassified samples rather than error due to $\tilde{Y}$. In additional tests (not shown here), the risk is estimated using the exact value of Y instead of $\tilde{Y}$ ($\tilde{R}_B = H(Y)\tilde{B}$ instead of $\tilde{R}$). The quantity $\tilde{R}_B$ is the risk prediction if the GP regression were exact. The use of R_B instead of $\tilde{R}$ leads to a relative error reduction around 3 % confirming that most of the error in the risk prediction comes from the classifier.

In Fig. 6.3, we study the influence of the batch size on the performance of the AL strategies. The results for three batch sizes and the two strategies AL-Heaviside and AL-probit are presented. To better observe the difference in performance due to the batch size, a large range of training sample size is explored. Due to computational cost constraints, the experiments for the batch size of 1 are only repeated twice and hence present a significant noise. While the asymptotic error for a given AL method seems to be independent of the batch size, in the first iterations of the

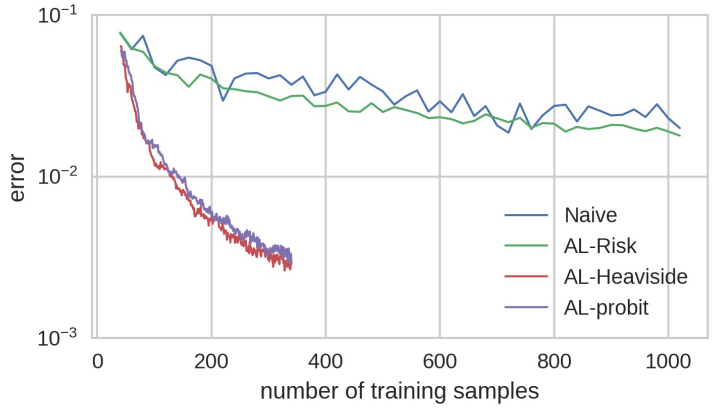


Figure 6.3: Evolution of the L2 error with the number of training points for the four different training approaches (2D test, single mode hazard function).To highlight the difference in performance, the AL-Risk and naive are tested with larger training sets than active learning strategies (up to 1000 samples *versus* 350 samples for the for the AL-probit and AL-Heaviside).

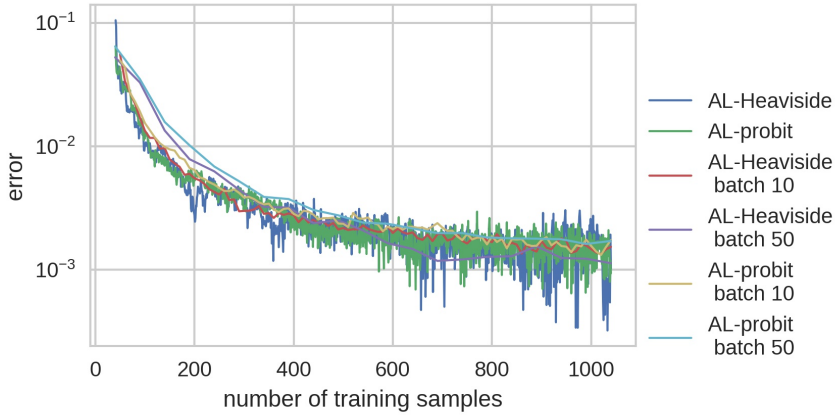


Figure 6.4: Evolution of the L2 error the the AL-probit and AL-Heaviside strategies for different batch sizes (1,10 or 50).

experiment, the 50 sample batch is outperformed by the other two methods. Beyond 700 samples, the 50 sample batch finally reaches the same level of error as the other two approaches. The difference between a batch of 1 and 10 are not clear due to the noise. It seems that the training strategies trained with batch size 10 are slightly outperformed between 40 and 200 samples. This test shows that the batch size choice is dictated by the number of training points to be added. As a rule of thumb, the batch size should not be too large such that the computational budget is depleted within a dozen iterations.

Training set distribution The AL advanced learning methods add training points according to a distribution G defined in 6.3. In figure 6.5, the training set is represented (red dots and blue stars) along with the function $Y \circ H$ in the background and the classifying boundary of function B in black. The training set represented by the red dots and the blue stars have been generated using the AL-probit strategy. One part of the training sample set is uniformly sampled (40 points) and represents the initial training set. The rest of the training set (100 training points)

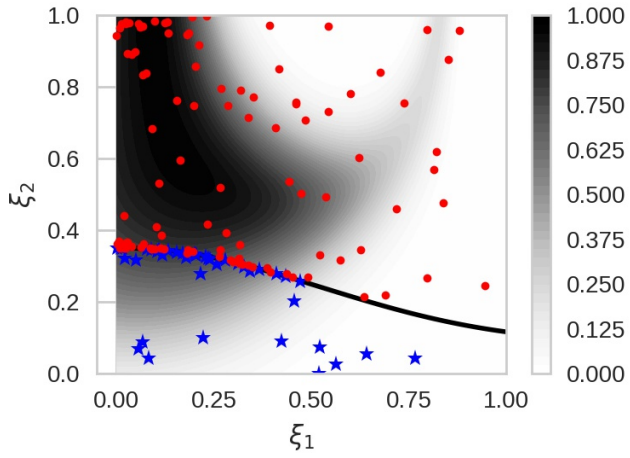


Figure 6.5: Training set distribution using the sigmoid training strategy. The red dots are surviving training points while the green stars are disintegrated samples. In the background is represented the hazard function composed with Y. The black line is the classifier boundary corresponding to $f(\xi_1, \xi_2) = 0$.

have been added in batches of size ten following the AL-probit strategy. The training points added by the active learner are distributed in regions near the boundary between the two classes defined by B and where the hazard function takes high values. This distribution was expected since most of the error is due to misclassification. The first term in Eq. (6.16) is driving the active learning strategy such that highly enriched regions are located near the boundary where $(p(1-p)$ is large)and where H is high. Note that the active learner does not try to learn precisely the classifier boundary where the hazard function is low. This is a key feature of our active learning criterion that focuses on risky regions where the prediction uncertainty is large.

Note the concentration of training points near the top left corner. In this region, the membership probability of the GP classifier is significantly lower than one and is coupled with high H values. Hence, the active learning proposes additional training samples in this region. This undesirable behavior is not due to the active learning criterion itself but rather to a general drawback of the GP for classification (and kernel methods in general) that have high predictive uncertainties in corners where the training point density is necessarily lower.

Predictive error In this work, we use the variance of $\tilde{R}$ as a surrogate of the actual L2 error between $\tilde{R}$ and R. In this paragraph, we assess the capability of the predictive variance to emulate the actual error. The predictive variance should be able to capture the relative contribution to error from the GP classifier and the GP regression. To verify that the contributions are correct we define:

$$E_c(\xi) = \left(H(\hat{Y}(\xi))\tilde{B}(\xi) - R(\xi)\right)^2 \tag{6.20}$$

that corresponds to the residual error when $\hat{Y}$ is replaced by $\tilde{Y}$ and

$$E_r(\xi) = \left(H(\tilde{Y}(\xi))B(\xi) - R(\xi)\right)^2 \tag{6.21}$$

that corresponds to the residual error when B is replaced by $\tilde{B}$. The quantity E_c measures the error due to the classifier while E_r measures the error due to the GP regression.

For E_c to be well defined, $\hat{Y}$ needs to be defined on Ω. For the purpose of the test, we define :

$$\hat{Y} : (\xi_1, \xi_2) \longrightarrow \left(\sin(4\xi_1^2)\xi_2^2 + \exp(-\xi_1)\sin(\xi_1)\exp(-\xi_1(1+\xi_1)), \xi_1^2 + \frac{1}{(1+\exp(3\xi_1\xi_2))} \right), \tag{6.22}$$

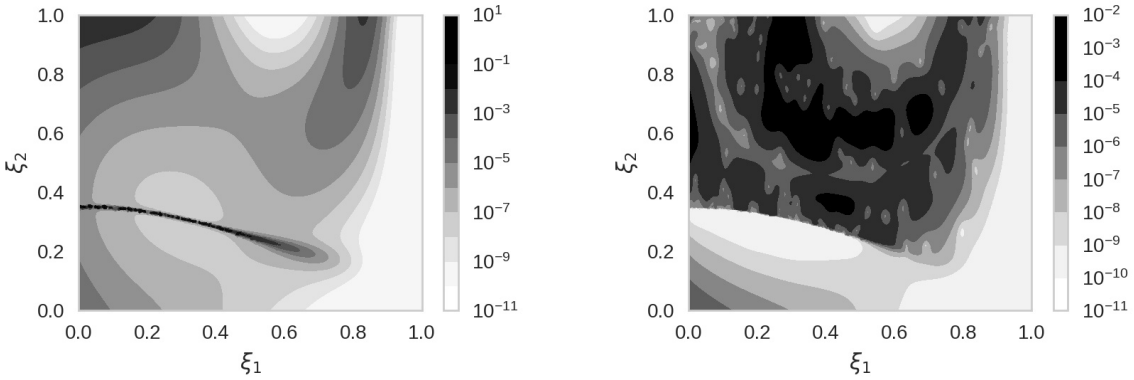
We have have that $\hat{Y} = Y$ on Ω_b. The two quantities are compared to the contributions to predictive variance (V_Y and V_B) in Figs. 6.6a, 6.6b, 6.6c and 6.6d. At first glance, the predictive errors ratio is in line with the actual error estimate: the error due to the classifier is much larger than the error due to the GP regression. Comparing 6.6a and 6.6c, we see that the actual error due to classification is concentrated on the boundary solely. The predictive contribution V_B predicts high error rates in this region too, but also around (0,1) and (0.8,1). Those regions correspond to high values of H and a membership probability significantly lower than one due to the lack of samples near the edge of the domain.

Concerning the contribution to error due to regression (Figs 6.6b and 6.6d), the most problematic disagreement is located around (0,0). In this region, the object always burns, but the membership probability remains strictly higher than zero, hence the term $V_Y = \mathbb{V}\left[H(\tilde{Y}(\xi))\right]\bar{\pi}(\xi)$ is not null. Moreover, the hazard function varies rapidly in that region and $\tilde{Y}$ has high predictive variance since Y is not defined in this region and there are therefore no training samples around. Consequently $\mathbb{V}\left[H(\tilde{Y}(\xi))\right]$ is large. This behavior is an undesirable consequence of trying to build a surrogate model of $\hat{R}$ (Eq. (6.2)) instead of R. Nevertheless, this error misprediction is extremely small compared to the variance contribution of $\tilde{B}$.

Uniform hazard function

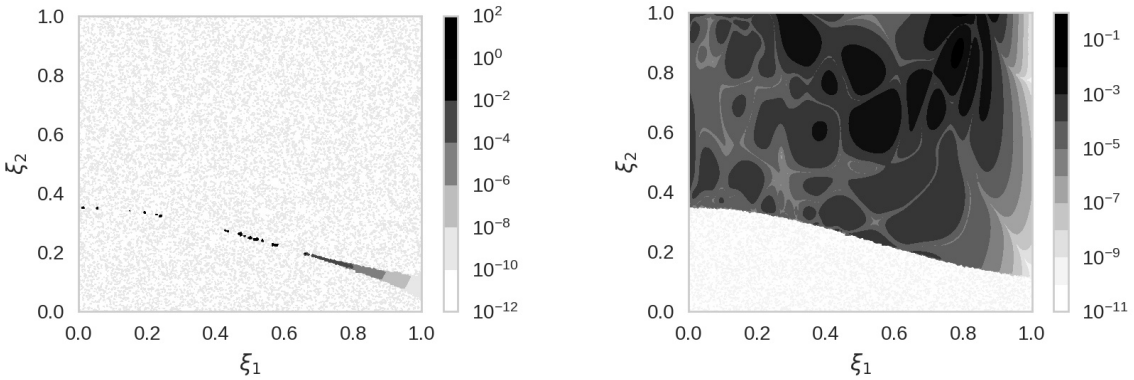
In this paragraph, we consider a uniform hazard function. In term of risk prediction, this case is a simple classification problem where only the misclassification error matters. The accuracy of $\tilde{Y}$ does not influence the result. The estimation of the predicted error $\mathbb{V}\left[\tilde{R}(\xi)\right]$ in Eq. (6.7) accounts for this characteristic as the contribution to variance from the regression term is zero due to $\nabla H = 0$. The risk variance reduces to the second term that is the classical uncertainty criterion used for classification weighted by the hazard function squared. In figure 6.9, the AL strategies AL-probit and AL-Heaviside are tested against the naive approach. In this case, since the risk is uniform, the naive and the AL-Risk approaches are the same and therefore AL-Risk is not reported. As for the unimodal case, the AL-probit and AL-Heaviside outperform the naive method while the AL-Heaviside approach seems to be performing a bit better than the AL-probit.

We also test the influence of the batch training size on the error performance for the active learning. Due to computational cost constraints, the batch size 1 case have been repeated only twice and hence features a significant noise. As for the unimodal case, the 50 batch size training strategies are outperformed for sample set smaller than 400 samples then perform as well as the other strategies trained with smaller batch sizes.



(a) Contribution from the classifier $\tilde{b}$ to predictive variance

(b) Contribution from the GP $\tilde{f}$ to predictive variance



(c) Residual error due to the classifier

(d) Residual error due to the regression

Figure 6.6: Comparison of the predicted contribution to error from the classifier (6.6a) and the GP regression (6.6b) with E_c (6.6c) and E_r (6.6d).

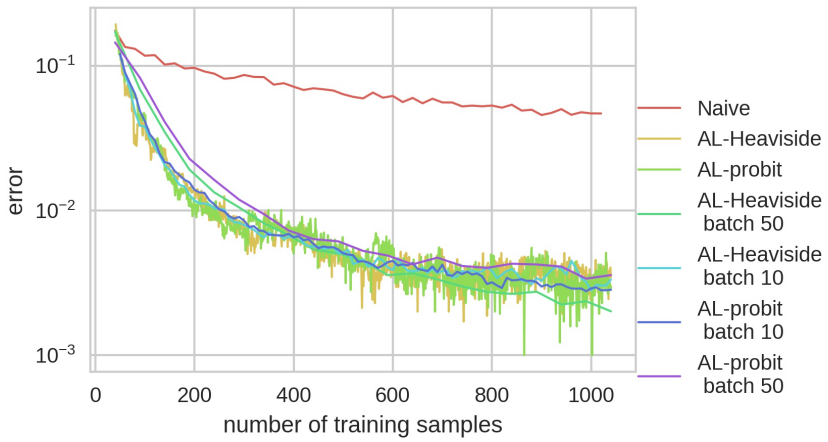


Figure 6.7: Evolution of the L2 error in the estimated with the number of training points for the three different training approaches (2D test, uniform hazard function)

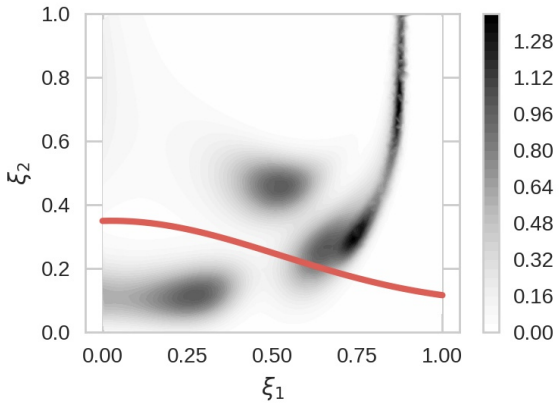


Figure 6.8: Representation of $H \circ Y$ and the survivability boundary associated with B (blue line)

Multi-modal hazard function

In this test, the hazard function is the sum of four Gaussian functions :

$$\begin{aligned} H : (y_1, y_2) \in \mathcal{Y} \longrightarrow & \exp\left(-(500(y_1-0.2)^2 + 10(y_1-0.2)(y_2-0.9) + 500(y_2-0.9)^2)\right) \\ & + \exp\left(-(100(y_1-0.2)^2 - 10(y_1-0.2)(y_2-0.8) + 100(y_2-0.9)^2)\right) \\ & + \exp\left(-(100(y_1-0.4)^2 - 10(y_1-0.4)(y_2-0.6) + 100(y_2-0.6)^2)\right) \\ & + \exp\left(-(200(y_1-0.1)^2 + 200(y_2-0.55)^2)\right). \end{aligned} \tag{6.23}$$

This multi-modal case is representative of a region with several highly populated areas (for instance cities). The function $H \circ Y$ is represented in Fig. 6.8 In this setup, the hazard function varies rapidly.
As for the previous cases, different training strategies are compared: the naive and the AL-Risk approaches against the AL-probit and AL-Heaviside approaches with batch size 10 and 50. In this case, the batch size 1 is omitted as the trends are identical to the previous cases. The results are summarized in Fig. 6.9. In this case, the AL-Risk and naive perform similarly and are outperformed by the AL advanced methods. As in the previous case, the AL-Heaviside training yields slightly better results than the AL-Probit training. In this case, the hazard function varies rapidly over $\mathcal{Y}$ and the active learner adds training samples where hazard function gradient is high or at the boundary of $\tilde{B}$ to reduce the error efficiently.

In Figs. 6.10a and 6.10b, the predictive variance used in the AL methods is compared to the actual error. The predictive variance tends to agree with the true error qualitatively, except for the lower left-hand corner where the true error is zero (the classifier accurately predicts complete disintegration), but the predictive variance is not negligible. Consequently, the active learning strategy tends to sample in this region although the actual error is already 0. This is obviously an undesirable feature of our strategy. On the other hand, the active learning strategies also efficiently sample additional points near the boundary where the risk is high (around (0.6,0.3)) and in regions where the risk varies rapidly (the line segment (0.7,0.3) to (0.9,1.)). Note that in this test, the overall contribution of $\tilde{Y}$ to the predicted error is not negligible compared to the classifier contribution, especially in regions where the hazard function varies rapidly.

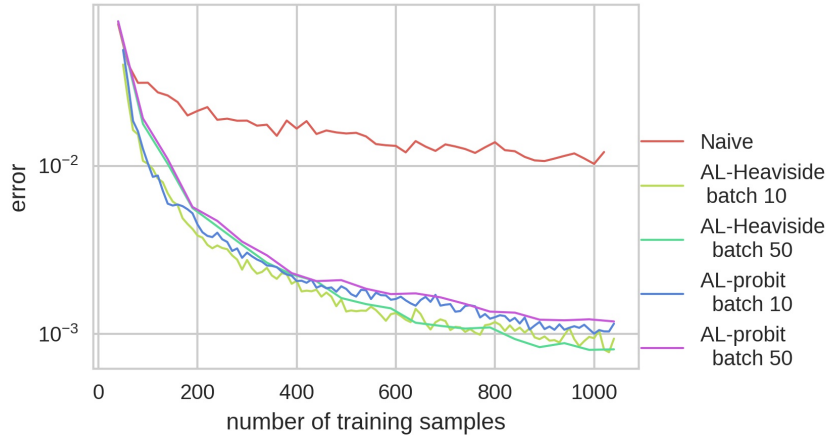
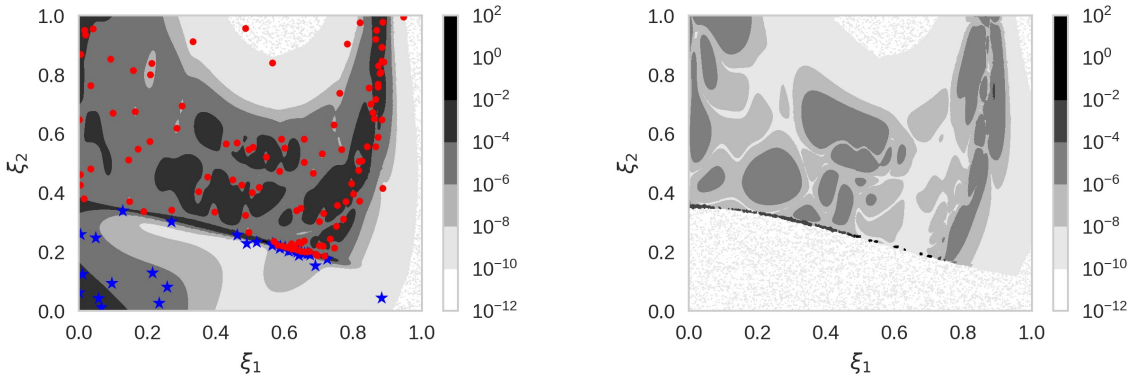


Figure 6.9: Evolution of the L2 error in the estimated with the number of training points for the three different training approaches (2D test, multi-modal hazard function)



(a) Predicted variance and training points

(b) True squared error

Figure 6.10: Comparison between the predicted error (6.10a) and the actual error (6.10b). In Fig. 6.10a the training samples are also represented. The red dots correspond to surviving debris while the blue stars correspond to non surviving debris.

6.4.2 Preliminary test for the impact prediction of a space object fragment

The primary objective of this work is to develop efficient surrogate model strategies to propagate uncertainties in a space object reentry problem. When a human-made space object reenters the Earth, the combined effect of the thermal and aerodynamic loads leads to its breakup and the generation of a fragment that may or may not reach the ground. In this test, we consider a debris that has just been released after fragmentation, and we wish to estimate the risk it represents for human assets. The fragment under consideration is an aluminum frame box issued after the fragmentation of an upper stage. The fragment weighs 80 kg. The objective of this test is to determine the risk associated with this fragment. We reuse the analysis from chapter 5. The initial upper stage (not modeled here) is originally on an equatorial Geo Transfer Orbit (GTO) such that the impact location is located in the Pacific ocean where the population density is null. Hence we introduce an artificial human population density (fake highly populated islands) for this test. The hazard function is represented on the globe in Fig. 6.11 along with a sample of impact locations.

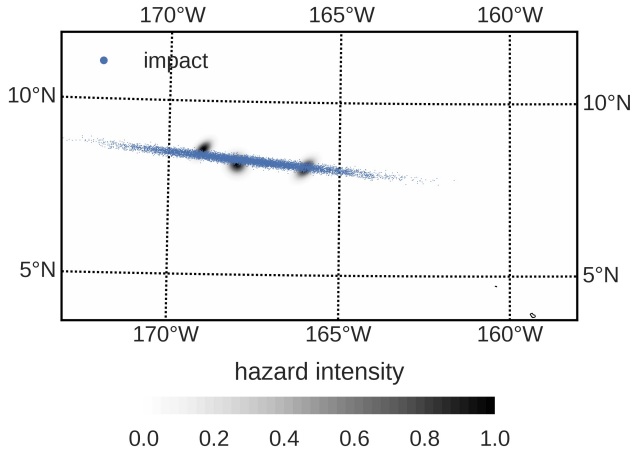


Figure 6.11: Representation of the impact region with an artificial hazard function. The blue dots correspond to impact locations.

We use the survivability solver of the simulator from section 3.2.1. It is the trajectory solver BL43 coupled with the thermal module ADRYANS V4. The thermal module solves for the aerodynamic heat flux and computes the resulting temperature increase and the ablation rate along the trajectory. It takes as inputs the initial flight conditions, the material characteristics and the object dimensions and weight. The solver is relatively expensive to run due to the strong coupling between the trajectory solver and the thermal module that is run at each time step. As described in the introduction such solver provides two types of information: whether the object survives upon reentry; if it does, the impact location. The hazard function is then used to compute the risk. In this test, it is defined as :

$$\begin{aligned} H(y_{\text{lat}}, y_{\text{lon}}) \mapsto & \exp\left(-15.((y_{\text{lat}}-8.2)^2+(y_{\text{lon}}+168)^2)\right) \\ & + \exp\left(-20((y_{\text{lat}}-8.1)^2-(y_{\text{lat}}-8.1)(y_{\text{lon}}+166)+(y_{\text{lon}}+166)^2)\right) \\ & + \exp\left(-25((y_{\text{lat}}-8.6)^2-(y_{\text{lat}}-8.6)(y_{\text{lon}}+169)+(y_{\text{lon}}+169)^2)\right) \end{aligned} \tag{6.24}$$

where y_{lat} and y_{lon} are the impact latitudes and longitudes.

In our problem, the fragment initial position, velocity and temperature are uncertain. The exact release conditions are incredibly complicated to predict since they depend on the fragmentation of the upper stage [37]. The initial position, and velocity uncertainties are modeled as a multivariate Gaussian vector with mean and covariance calibrated on the results of Chapter 5. The initial fragment temperature also depends on the flight history of the original upper stage which is unknown. Hence, the initial fragment temperature is modeled as a uniform random variable. The material characteristics such as fusion temperature, density, and fusion enthalpy are Gaussian random variables obtained from an experimental measurement campaign. The emissivity is modeled as a uniform random variable because it depends on the surface oxidation that is uncertain. Overall there are 11 uncertainties to be propagated that are summarized in table 6.1.

The initial flight conditions are samples from a multivariate Gaussian vector Φ with mean

Quantity	Distribution	
Initial altitude	multivariate Gaussian (see eqs. 6.25 and 6.26)	$[m]$
Initial longitude	multivariate Gaussian (see eqs. 6.25 and 6.26)	$[deg]$
Initial latitude	multivariate Gaussian (see eqs. 6.25 and 6.26)	$[deg]$
Initial velocity norm	multivariate Gaussian (see eqs. 6.25 and 6.26)	$[m/s]$
Initial slope	multivariate Gaussian (see eqs. 6.25 and 6.26)	$[deg]$
Initial bearing	multivariate Gaussian (see eqs. 6.25 and 6.26)	$[deg]$
Initial object temperature	$\mathcal{U}(400, 700)$	$[K]$
Aluminum fusion temperature	$\mathcal{N}(873., 0.4)$	$[K]$
Aluminum fusion enthalpy	$\mathcal{N}(350, 3)$	$[kJ/kg]$
Aluminum density	$\mathcal{N}(2800., 10.)$	$[kg/m^3]$
Aluminum emissivity	$\mathcal{U}(0.3, 0.8)$	

Table 6.1: List of uncertainties associated with the survivability prediction of a space debris

and covariance:

$$\mu = \begin{pmatrix} 63000 \\ -170 \\ 8.5 \\ 9300 \\ -2.9 \\ 95 \end{pmatrix} \tag{6.25}$$

$$\Sigma = \begin{pmatrix} 1.50 \times 10^7 & 2.62 \times 10^2 & -40.5 & 9.26 \times 10^5 & 3.37 \times 10^2 & 57.4 \\ 2.62 \times 10^2 & 2.00 & -0.18 & -82.3 & 0.84 & 0.28 \\ -40.5 & -0.18 & 2.00 \times 10^{-2} & 7.37 & -8.60 \times 10^{-2} & -3.15 \times 10^{-2} \\ 9.26 \times 10^5 & -82.3 & 7.37 & 6.7 \times 10^4 & -23.8 & -12.0 \\ 3.37 \times 10^2 & 0.84 & -8.6 \times 10^{-2} & -23.8 & 0.4 & 0.13 \\ 57.4 & 0.28 & -3.15 \times 10^{-2} & -12.0 & 0.13 & 5.0 \times 10^{-2} \end{pmatrix} \tag{6.26}$$

We use the approach presented in this work to build and train a surrogate model of the solver efficiently and get an estimate of the risk. The surrogate model is constructed following the strategy described in Section 6.3 where $\tilde{B}$ is built with a probit function and an anisotropic square exponential covariance function. The GP $\tilde{Y}$ is also constructed using a zero mean Gaussian Process with a square exponential anisotropic covariance function. The surrogate model is initially trained with 500 samples sampled according to the input distributions. The active learning strategy developed in section 6.3 is used to enrich the training set with 960 additional samples. As the solver can be run in parallel, the training samples are added by batches of 48 samples. For comparison, the results obtained with randomly generated samples are presented. The error evolution with the number of samples is shown in Fig. 6.12. The error represented is the L1 norm computed using 100 000 validation samples. The L1 norm is normalized by the mean risk such that a naive predictor that would predict a null risk would have an error of 1. The results clearly demonstrate the advantage of using an active learning strategy to improve the surrogate model. At 1500 training samples, the AL strategy is one order of magnitude more accurate than the random training strategy and present a steeper converging rate.

As mentioned in the introduction, we are primarily interested in computing statistics of the risk distribution such as the probability that the risk be above 10^{-5} (maximum level allowed by

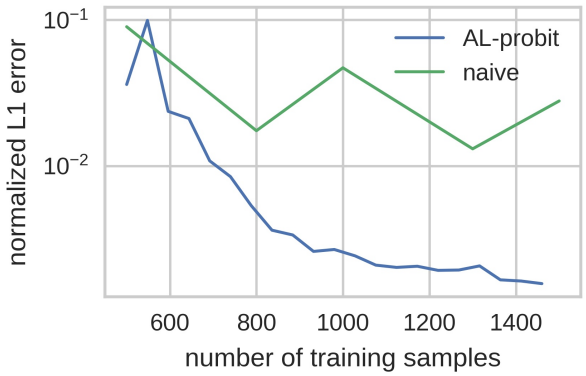


Figure 6.12: Evolution of the L1 norm error for our active learning method and the random training one.

Quantity	Reference	Active Learning	Naive	MC error
$P(R > 10^{-5})$	0.18767	0.18772	0.18957	0.004
$P(R > 0)$	0.25390	0.25394	0.25667	0.004

Table 6.2: Comparison of the random and active learning methods when computing risk probabilities compared to a reference. Note that the same Monte Carlo samples were used to estimate the probabilities.

the mitigation guidelines) as the probability that the risk be non zero. In table 6.2, the numerical values for $P(R > 10^{-5})$ and $P(R > 0)$ are estimated using the 100 000 validation samples as a reference. Using the same input samples, the quantities $P(R > 10^{-5})$ and $P(R > 0)$ are estimated with two surrogate models. The first one is trained with the 1460 samples proposed by the active learning method while the other one is constructed with 1500 points randomly sampled from the input distribution. The Monte Carlo error is given for reference. However, since the same samples are used to compute the risk statistics for the three strategies, the difference of results between the three methods does not come from the MC statistical noise. The results show that the AL training set outperforms the naive training set by one order of magnitude. Note however that the results obtained with the naive training strategy remain decent as the error remains below the MC estimation error obtained with 100 000 samples.

6.5 Conclusion

In this chapter, we presented two original contributions for computing the statistical distribution of the expected number of victims in the context of the space object reentry prediction under uncertainties. The first one is the construction of a surrogate model of a discontinuous risk function using a classifier and a regression. The second contribution is the derivation of active learning strategies based on the surrogate model predictive variance. The combination of the surrogate model coupled with the active learning methods showed to be efficient on analytical test cases and numerical reentry prediction solver with a large number of uncertainties. The precision gain of using an active learning method is around one order of magnitude compared to a random training method and therefore justifies the use in a realistic reentry problem. In the next Chapter, we will use this formulation to efficiently build a surrogate model of the survivability

solver in the reentry simulator developed in Chapter 3.

Chapter 7

Risk Assessment of the Reentry of a Rocket Upper Stage

In this Chapter we study the controlled reentry of an upper stage under uncertainty with a particular focus on the survivability of the fragments released after breakup. The propagation of the uncertainties through the entire simulator relies on the coupling of the SoGP framework developed in Chapter 4 and the surrogate model strategy presented in Chapter 6. This chapter includes the calculation of the survival probability for each fragment, the impact location, the expected risk, and the total casualty area. Compared to the study performed in Chapter 5, where the analysis was limited to the upper stage breakup prediction, we simulate the entire reentry under uncertainty, including the fragment trajectories until impact or complete demise.
The main contributions of this chapter are:

- *A full-scale application of the SoGP framework coupled with the survivability surrogate model;*
- *The rigorous and computationally efficient propagation of the uncertainties in a reentry event from the initial orbital conditions to the ground impacts of the fragments. We use the active learning strategies developed in the previous chapters. We are able to tailor the computational budget depending on the required level of accuracy.*
- *Computation of the statistical distributions of the on-ground risk metrics and rigorous sensitivity analysis to identify predominant uncertainties.*

7.1 Problem definition

We consider the controlled reentry of an upper stage from a GTO orbit. When the deorbited upper stage falls back on Earth, the aerothermal loads lead to its breakup (cf. Chapter 5). As the object breaks up, fragments are released. The fragments are released at hypersonic velocities ($M > 5$) at an altitude ranging from 50 to 80 km. As the fragments fall back, the atmospheric density increases and induces an increase in the heat flux received by the fragments. Some fragments burn up into the atmosphere due to the extreme thermal load. Others, typically made of titanium or composite materials survive. Around 20 km altitude, most fragments are flying in the subsonic regime so that the convective heat flux received by the fragment is null. The object may even transfer heat to the flow if the object wall temperature is high. The surviving fragments usually impact the ground at their free-fall velocity (around several dozen meters per

second).

For this specific case chosen by AG, the expected number of casualty is null since the reentry is controlled and designed to minimize the on-ground risk. Hence, for this study, the QoI are total casualty area that characterizes the potential threat of a reentry independently of the reentry location, the survival probability for each fragment and the impact distribution. Finally, we also perform a sensitivity analysis of the casualty area uncertainty to identify the most influential input uncertainties.

Object characteristics

The upper stage characteristics remain unchanged with respect to Chapter 5. In our breakup model, the object fragments to be released upon breakup are defined by the user. The list of fragments is the same as for the preliminary tests run in Chapter 3 and is reported in table 7.1. Note that there are 29 types of fragments but certain types are repeated (there are 4 pressure spheres for instance) and so the total number of fragment is 100. Due to the survivability model limitations, the fragment shapes are constrained to a limited choice of predefined shape for which the aerothermal loads and aerodynamic forces can be computed. For each type of fragments, the user has to provide the dimensions, the shape, the attitude motion, and the material. Note that some fragments share the same characteristics.

The user also has to provide the fragments release temperature T_{init}. In this numerical experiment, all the fragments are assumed to have the same release temperature. The fragment characteristics (shape, material, and mass) are based on a priori structure analysis of a generic upper stage performed by engineers at ArianeGroup. The fragments are composed of four types of material: aluminum, stainless steel, titanium, and Inconel. Over 75 % of the total fragment mass is aluminum which is the most fragile material.

Id	Number	Name
1	4	Payload adapter rings
2	1	Payload adapter
3	4	Cone
4	4	Equipments plate
5	3	Equipment bay structure 1
6	3	Equipment bay structure 2
7	4	Payload adapter Ring
8	4	Equipment bay structure 3
9	4	LH2 tank upper skirt 1
10	4	LH2 tank upper skirt 2
11	8	LH2 upper dome
12	4	Equipment bay structure 4
13	3	LH2 tank cylinder
14	4	LH2 tank lower skirt
15	4	LH2 tank ring
16	8	LH2 tank lower dome
17	3	Lox tank lower dome
18	3	Lox tank upper dome
19	2	Cold Gas Reacting System
20	8	Engine Thrust Frame 1
21	1	Engine Thrust Frame 2
22	1	Engine Thrust Frame 3
23	3	LH2 feedlines
24	3	LOX feedlines
25	4	Helium spheres
26	1	Combustion chamber
27	3	Power unit
28	1	Turbopump 1
29	1	Turbopump 2

Table 7.1: List of fragments

7.2 Numerical model

We use the model described in section 3.2.1 to simulate the object reentry. The prediction includes the orbit propagation after the deorbiting boost, calculation of the pre-breakup trajectory and the derivation of the breakup conditions. At breakup time, a set of pre-computed fragments are released at once. The fragment initial velocity and position are given by the upper stage flight conditions at breakup. After breakup, the trajectory, temperature and mass evolution are computed for each fragment individually. The survivability solver calculates the trajectory and mass evolution to determine whether the object survives to reentry (i.e., reaches the ground) and the impact location if any. Hence, the survivability solver and the risk estimation solver have to be run for each fragment. In Fig. 7.1, the system of solvers in use is represented. For clarity, the survivability solver is repeated for each fragment to highlight the fact that each fragment requires one call from the survivability solver.

Based on the survivability output, the casualty risk defined as the expected number of victims can be computed using Eq. 2.8 from Chapter 2. As the object falls back on the ocean, the expected number of casualty is null. While we are still interested in verifying that the expected number of vitims remains negligible under the uncertainties considered, we are also interested in computing the casualty area. The global casualty area is defined as :

$$A = \sum_{i}^{n} A_{i,k} \mathbb{1}_{i,survi} \tag{7.1}$$

where $\mathbb{1}_{i,survi}$ equals 1 if the fragment i survives, 0 otherwise.

The casualty area quantifies the potential damage the fragments could cause if they were to fall in a densely populated area.

In addition to the casualty risk and the casualty area, we also compute the impact positions. The impact positions are given in terms of longitudes and latitudes by the survivability solver. The true distances and area are computed using the elliptical Earth WGS84 model (World Geodetic System 1984) and numerical routines from [210] to compute geodesic areas and distances.

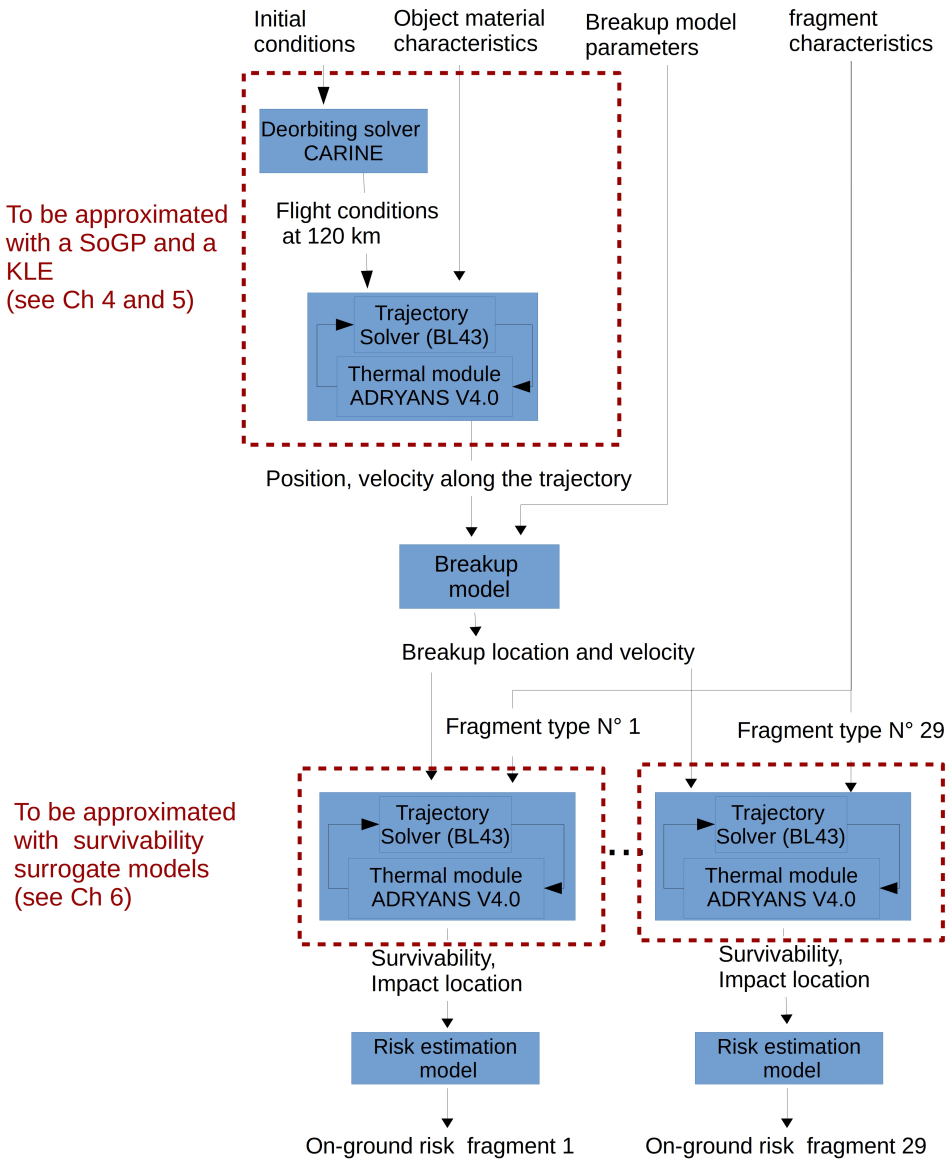


Figure 7.1: Numerical model for predicting the reentry of an upper stage from a GTO orbit.

7.3 Uncertainty characterization

The objective of this chapter is to perform an analysis of the model uncertainties. In this section, we model a large number (38) of uncertainties to perform a complete reentry simulation. The uncertainty modeling presented in Chapter 5 is employed in this Chapter. In particular, we consider as uncertain quantities the initial conditions (initial orbit characteristics, the deorbiting boost conditions), the atmosphere model parameters, the object material characteristics, the breakup model parameters, the fragment material characteristics, and the fragment release temperature. The uncertainty distributions are summarized in table 7.2. In addition to the uncertainties already considered in Chapter 5, we consider the fragment uncertain material characteristics and the fragment release temperature. In the rest of the section, we detail the

modeling of those additional uncertainties.

Material properties

For each material, we consider the emissivity, the density, the fusion temperature, and fusion enthalpy. The emissivity is modeled as a uniform distribution to account for the oxidation layer created by the high enthalpy reacting flow. For titanium, we define the range as $[0.3, 0.8]$, based on experimental results reported in [59]. For Inconel, the measurement campaign under high enthalpy flow presented in [91] reports emissivity variations between 0.2 and 0.8 and even 0.9. In this study, we consider that the Inconel emissivity varies between $[0.2, 0.8]$. For aluminum, the same range of variation as for the upper stage shell is selected: $[0.3, 0.8]$. For stainless steel, the emissivity is arbitrarily chosen between 0.6 and 0.8. The uncertainties associated with the other material quantities are modeled as Gaussian random variables since they are measured quantities.

Note that the aluminum fragment properties and the aluminum upper stage are modeled with two sets of independent uncertainties. This approach is justified for two reasons. First, the sensitivity analysis indices we use in this work (Sobol indices) are only valid for independent random variables. By using two separate sets of uncertainties to model the upper stage material characteristics and the aluminum fragment uncertainties, we can identify the contributions of each set of properties to the casualty area variance. Secondly, the uncertainty in the material is partially due to the manufacturing process, different for each fragment, and the flow exposure which is different for the shell and the fragments.

Breakup model parameter T_{init}

The breakup model uses a simplified representation of the upper stage. The object is divided into an aluminum shell and the internal components. Before the breakup, the shell absorbs all the energy transferred by the flow. The shell is assumed to have a uniform temperature. As the internal components are not modeled in the thermal response, their release temperature is unknown. Their release temperature depends on their exposure to the flow (the internal tank component are not as exposed as the payload adapters for instance), the thermal conductivity in the upper stage and internal radiative heat flux. All those aspects are not accounted for in the thermal model in use. Hence we model the release temperature uncertainty as a uniform random variable varying between the free stream temperature 300 K and 600 K. In theory, each fragment should have a different release temperature. To maintain the number of uncertainties to a reasonable level, we consider a single release temperature identical for all fragments.

Discussion

In addition to the uncertainties already included in the breakup analysis from chapter 5, we also consider the uncertainties in the material characteristics of the fragments and the release temperature. While we already model 38 uncertainties for this study, many other uncertainties could have been included. The fragment characteristics (shape, size, and number) could be modeled as uncertain variables. In theory, they depend on the breakup characteristics (low energy or high energy, thermal or mechanical breakup, etc.), while in our model the fragment characteristics are fixed and based on *a priori* considerations. The aerodynamic and thermal loads are computed using simple models, and the trajectory is solved with a three DoF model with a predefined attitude motion. Including those sources of uncertainty is beyond the scope but would undoubtedly bring an interesting insight into the problem.

Variable	Description	Category	Distribution
ρ_M	material density	OMC	$\mathcal{N}(2800., 10.)$ $[kg/m^3]$
ϵ	material emissivity	OMC	$\mathcal{U}(0.3, 0.8)$
T_{fus}	fusion temperature	OMC	$\mathcal{N}(873., 0.4)$ $[K]$
H_{fus}	fusion enthalpy	OMC	$\mathcal{N}(350, 3)$ $[kJ/kg]$
A_{boost}	boost amplitude	IC	$\mathcal{U}(62, 67)$
t_{boost}	boost time	IC	$\mathcal{U}(0, 5)$ $[s]$
α_{boost}	boost inclination	IC	$\mathcal{U}(178, 182)$ $[deg]$
β_{boost}	boost bearing	IC	$\mathcal{U}(-90, 90)$ $[deg]$
apo	initial orbit apogee	IC	$\mathcal{U}(35e6, 35e6 + 35e3)$ $[m]$
per	initial orbit perigee	IC	$\mathcal{U}(254000, 256540)$ $[m]$
inc	initial orbit inclination	IC	$\mathcal{U}(10, 10.1)$ $[deg]$
anl	initial orbit ascending node longitude	IC	$\mathcal{U}(-135, -134.9)$ $[deg]$
pa	initial orbit perigee argument	IC	$\mathcal{U}(130, 130.01)$ $[deg]$
tan	initial orbit true anomaly	IC	$\mathcal{U}(43, 43.01)$ $[deg]$
atmoday	reentry day	AP	$\mathcal{U}[1, 365]$ $[day]$
atmohour	reentry hour	AP	$\mathcal{U}[0, 23]$ $[hour]$
solarf	solar flux	AP	$\mathcal{U}[65, 240]$
ag	magnetic index	AP	$\mathcal{U}[2, 75]$
T_{frag}	fragmentation temperature	BP	$\mathcal{U}[400, 700]$ $[K]$
p_{abl}	ablation percentage	BP	$\mathcal{U}[0.7, 0.9]$
T_{init}	fragment release temperature	BP	$\mathcal{U}(300, 600)$ $[K]$
ρ_{AL}	aluminum density	FC	$\mathcal{N}(2800., 10.)$ $[kg/m^3]$
ϵ_{AL}	aluminum emissivity	FC	$\mathcal{U}(0.3, 0.8)$
$T_{fus,AL}$	aluminum fusion temperature	FC	$\mathcal{N}(873., 0.4)$ $[K]$
$H_{fus,AL}$	aluminum fusion enthalpy	FC	$\mathcal{N}(350, 3)$ $[kJ/kg]$
ρ_{ST}	stainless steel density	FC	$\mathcal{N}(2800., 10.)$ $[kg/m^3]$
ϵ_{ST}	stainless steel emissivity	FC	$\mathcal{U}(0.3, 0.8)$
$T_{fus,ST}$	stainless steel fusion temperature	FC	$\mathcal{N}(873., 0.4)$ $[K]$
$H_{fus,ST}$	stainless steel fusion enthalpy	FC	$\mathcal{N}(350, 3)$ $[kJ/kg]$
ρ_{TI}	titanium density	FC	$\mathcal{N}(2800., 10.)$ $[kg/m^3]$
ϵ_{TI}	titanium emissivity	FC	$\mathcal{U}(0.3, 0.8)$
$T_{fus,TI}$	titanium fusion temperature	FC	$\mathcal{N}(873., 0.4)$ $[K]$
$H_{fus,TI}$	titanium fusion enthalpy	FC	$\mathcal{N}(350, 3)$ $[kJ/kg]$
ρ_{IN}	inconel density	FC	$\mathcal{N}(2800., 10.)$ $[kg/m^3]$
ϵ_{IN}	inconel emissivity	FC	$\mathcal{U}(0.3, 0.8)$
$T_{fus,IN}$	inconel fusion temperature	FC	$\mathcal{N}(873., 0.4)$ $[K]$
$H_{fus,IN}$	inconel fusion enthalpy	FC	$\mathcal{N}(350, 3)$ $[kJ/kg]$

Table 7.2: List of uncertainties with their probability distribution. In the last but one column, the uncertainty category is reported. IC : Initial Conditions, OMC : Object Material Characteristics, AP : atmosphere parameters, FC : Fragment Characteristics. The category corresponds to labels used in Figs 7.1 and 7.2.

When computing the trajectory of the upper stage before fragmentation, the atmosphere model parameters are considered as uncertain, but after fragmentation, we neglect the influence of their uncertainties. This simplification acceptable because the variability of the atmosphere properties (density, temperature, and pressure) is significantly reduced below 80 km [3]. On the other hand, when the object velocity is subsonic, the wind effects may become noticeable. In this study, the influence of winds is not considered.

7.4 Uncertainty propagation strategy

In this section, we build a global surrogate model used to propagate the uncertainties described in 7.3 through the SoS describing the whole reentry and presented in 7.2. We also adapt the active learning strategy developed in Chapter 6 to improve the accuracy of the predictions.

7.4.1 Surrogate model construction

We use the methods developed in Chapter 4 and Chapter 6 for constructing the surrogate model of this SoS. The global surrogate model of the entire SoS is represented in Fig. 7.2. It is composed of several surrogate models constructed independently. The first one is a SoGP composed with a KLE that is used to predict the object breakup. The other surrogate models approximate the survivability solver for each fragment. One surrogate model per fragment is constructed. In the rest of this section, we detail the surrogate construction of the SoGP and the survivability surrogate models.

For the upstream part of the SoS involving the breakup prediction, we reuse the procedures presented in Chapters 4 and 5. A surrogate model composed of a SoGP and a KLE approximates the deorbiting solver and the trajectory solver coupled with ADRYANS V4. It takes as inputs the initial flight conditions (initial orbit and deorbiting characteristics), the atmosphere model parameters and the object material characteristics, and it returns the object trajectory (position and velocity) between t_{init} and t_{end} that is then used by the breakup solver. The surrogate model construction approach is identical to the one presented in Chapter 5. See Section 5.3 for the implementation details. The surrogate model relies on a truncated Karhunen-Loéve Expansion (KLE) to represent the object trajectory on a reduced basis. A SoGP is then built to approximate the mapping between the uncertain inputs ξ and the coefficients of the KLE. The KLE is constructed using the procedure presented in section 4.1.4 using 600 trajectories. The SoGP is also constructed using the same 600 simulator runs. The training sample set is generated with a LHS sample.

For the downstream part of SoS including the risk estimation, each type of fragment (characterized by the shape, size, mass, and material, defined in the previous section) is treated independently in the simulator. Therefore, we construct a survivability surrogate model for each. In total, 29 types of fragments have to be considered. The surrogate model construction follows the strategy proposed in Chapter 6.

The input variables of the survivability solver are: the breakup flight conditions (position and velocity) which are intermediate variables computed by the breakup solver, the fragment characteristics (material emissivity, density, fusion temperature, fusion enthalpy), and the initial fragment temperature that are global input variables.

The survivability solver is approximated with a GP classifier $\tilde{B}_i$ and a GP regression $\tilde{Y}_i$ for each type fragment i. The classifier assesses the survivability of the fragment while the GP regression computes the impact location if the object survives. In total, each survivability

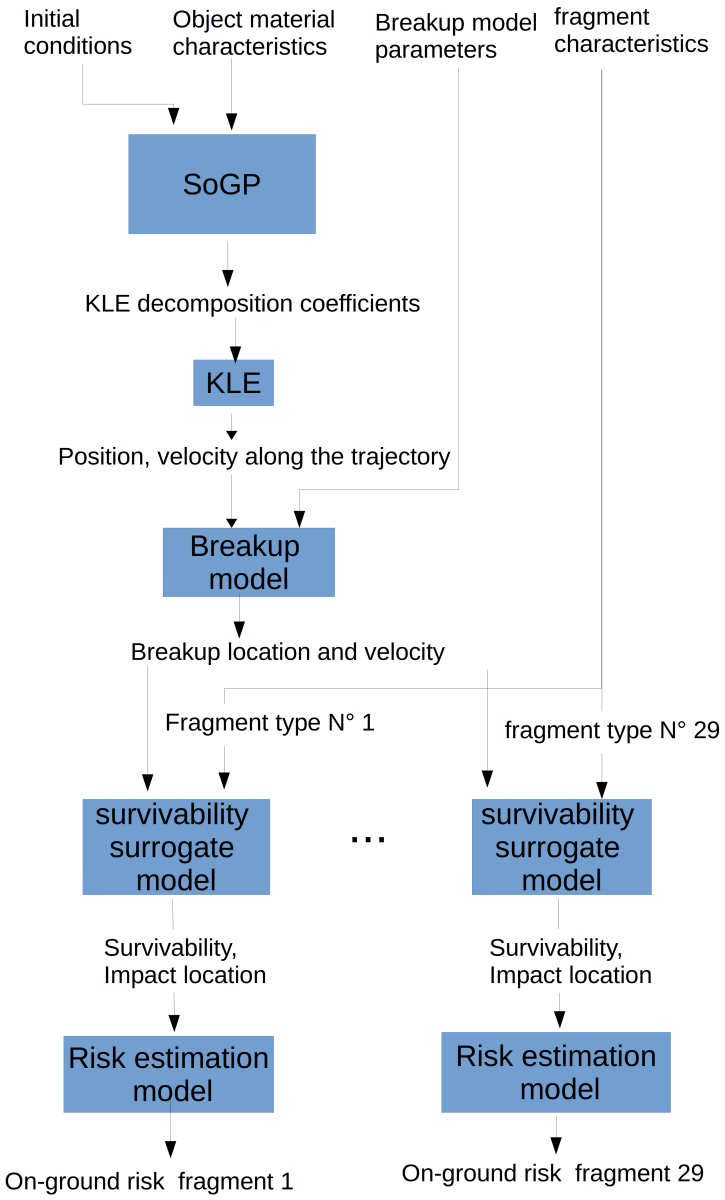


Figure 7.2: System of surrogate models used to build a global approximation of the simulator

surrogate model has 11 input dimensions. The initial training set contains 600 randomly sampled points. In the intermediate variable space (i.e., the breakup flight conditions), the training points are randomly sampled from a large dataset of points propagated by the complex SoGP-KLE and breakup model. For the global input variables, a LHS strategy is used to generate the training samples.

Out of the 29 types of fragments considered, a significant subset always burns up in the atmosphere for all 600 training samples, and only one always reaches the ground. Just for six fragments, we observe the two regimes: complete demise or ground impact. The rest either always burnt up completely or survived (for all the 600 training samples). For the fragments

that disintegrated in the atmosphere for all the 600 training samples, the risk is assumed to be 0 for any input conditions, and therefore they are excluded from our risk estimation study. This approach is not strictly conservative since having all the 600 samples disintegrated does not ensure that the fragment always disintegrate for any input condition. Actually, for most fragments, there is a non zero (although extremely low) probability that the object reaches the ground. Computing this probability requires dedicated uncertainty quantification techniques (see for instance [168, 211, 212, 213, 214]) that are out of the scope of this study.

7.4.2 Active Learning

We propose to enrich the training sample of the survivability surrogate model to improve the prediction accuracy using the training strategy proposed in Chapter 6. We enrich each training plans with an additional 432 samples.

The active learning strategy proposed in Chapter 6 is adapted to our application. There are two differences with the framework proposed in Chapter 6. First, the upper stage fragments fall into the Pacific ocean where the population density is null according to the population model in use. In Chapter 6, we used the risk predictive variance to propose additional training samples. The predicted risk variance for fragment k is :

$$\mathbb{V}\left[Rs_k\right] = \mathbb{V}\left[H\left(\tilde{Y}_k(\xi)\right)\right] p_k(\xi) + \mathbb{E}\left[H(\tilde{Y}_k(\xi))\right]^2 (1 - p_k(\xi))p_k(\xi). \tag{7.2}$$

Then additional training samples were drawn from a distribution X on Ω with pdf proportional to $\mathbb{V}\left[Rs_k\right]$. Recall that H is the hazard function. Since the population density is null, $H = 0$ in our case, therefore this criterion becomes inadequate.

Instead we propose to improve our casualty area prediction given by

$$\tilde{A}_k = A_{c,k} \mathbb{1}_{\tilde{B}_k(\xi)=1}(\xi), \tag{7.3}$$

where $\mathbb{1}_{\tilde{B}_k(\xi)=1}(\xi)$ the indicator function of the set $\{\xi \in \Omega | \tilde{B}_k(\xi) = 1\}$. We propose to adapt the active learning strategy proposed in Chapter 6 to the predictive variance associated with the casualty area prediction is :

$$V_k = A_{c,k}^2 (1 - p_k(\xi))p_k(\xi) \tag{7.4}$$

This formulation is actually equivalent to the uniform density population case discussed in Chapter 6 where the active learner focuses on the classification error. This feature is desirable as the casualty area prediction depends on the fragment survivability only. It is zero if the fragments burns up and $A_{c,k}$ if the object reaches the ground. Note that we neglect the effects of ablation on the fragment casualty area in [64].

The second difference with Chapter 6 is that we consider several fragments at once in this study whereas in Chapter 6 we only considered single fragment case. In the case of several fragment, the total casualty area is defined as

$$\tilde{A} = \sum_{k}^{N} A_{c,k} \mathbb{1}_{\tilde{B}_k(\xi)=1}(\xi) \tag{7.5}$$

Since there is one surrogate model for each fragment, the predictive variance associated with the total casualty area is defined as:

$$V = \sum_{i}^{N} V_i \tag{7.6}$$

Hence three possible variations of the training strategy presented in Chapter 6 are possible :

- The next training sample ξ^* is sampled from a distribution with pdf proportional to $V = \sum_i^N V_i$ and all the $\tilde{R}_i$ are trained at location ξ^*. This strategy requires one evaluation of the survivability solver for each fragment.

- For each fragment i, the next training sample ξ_i^* is sampled from a distribution with pdf proportional to V_i and the surrogate model $\tilde{R}_i$ is trained at location ξ_i^*. This strategy requires one evaluation of the survivability solver for each fragment.

- We first select the surrogate model we wish to train by sampling from a discrete distribution valued in $[1, 2, \ldots, n]$ with weights $V_1/V, V_2/V, \ldots V_n/V$. The sampled index is denoted j. We then sample a new training sample ξ_j^* from a distribution with pdf proportional to V_j and the surrogate model $\tilde{R}_j$ is trained at location ξ_j^*. In this approach, only one surrogate model is trained at each iteration.

The third approach seems to be the most efficient, especially if one fragment is significantly larger than the others. In that case, the active learner will more likely add a training sample and improve the prediction of the fragment that has the largest influence on the total casualty area prediction error and will avoid spending training samples on non-influential fragments.

The first and second approaches enrich the training sets of all the survivability surrogate models regardless of the contribution of each fragment to the total casualty area predictive variance. However, the second approach seems more advantageous as it can select different training point location based on each surrogate model predictive uncertainty. In this study, the second approach is selected because it proposes training samples adapted to each surrogate model and we we wish to test the efficiency active learning strategy on each fragment. Furthermore, the additional training samples are added by batches of 48 samples.

7.5 Results

In this section, we collect the main results of this study. We first assess the accuracy of the global surrogate model composed of a SoGP and the survivability surrogate models. We then present the results in terms of survivability probability of each fragment, casualty area, and impact zone.

7.5.1 Error assessment

In this section, we quantify the approximation error of the surrogate model constructed. The validation error is computed using a validation set of 2400 samples denoted $\mathcal{V} = \{\xi_{v,1}, \ldots, \xi_{v,n}\}$ different from the training set used to train the surrogate model.

For the SoGP and the KLE used to emulate the deorbiting solver and the trajectory coupled with the thermal solver, the error analysis is similar to the results presented in Chapter 5. The l_2 error is around 10^{-3} for the breakup conditions (position and velocity). The detailed errors are reported in table 7.3. The l_2 error is normalized by the standard deviation of the breakup condition. This error comes from the KLE truncation, the approximate estimation of

	altitude	longitude	latitude	velocity	slope	bearing
l_2 normalized error	0.003	0.002	0.003	0.003	0.002	0.002

Table 7.3: Surrogate model error at the breakup conditions. The mean square error is computed using 2400 samples. It normalized by the standard deviation. This error includes the error induced by the SoGP and the KLE.

the eigenvalues and the eigenfunction of the decomposition, and the error induced by the SoGP when predicting the KLE coefficients based on the uncertain inputs.

To assess the precision of the survivability surrogate model, we define two types of errors: the survivability assessment error and impact location error. The survivability assessment error defined for fragment j is:

$$E_{sa,j} = \frac{1}{n}\sum_{i=1}^{n} |B_j(\xi_{v,1}) - \tilde{B}_j(\xi_{v,1})| \tag{7.7}$$

where B_j returns 1 if fragment j survives, 0 otherwise and $\tilde{B}_j$ returns 1 if the survivability surrogate model composed with the SoGP predict that the object survives, 0 otherwise. Furthermore, we define the sample set $\mathcal{I}_j = \{\xi_{v,i} \in \mathcal{V} | \tilde{B}_j(\xi_{v,i}) = 0 \text{ and } B_j(\xi_{v,i}) = 0\}$ and the impact location error :

$$E_{il,j} = \frac{1}{|\mathcal{I}|}\sum_{\xi \in \mathcal{I}} |f(\xi) - \tilde{f}(\xi)| \tag{7.8}$$

The impact location error is defined for surviving debris solely.

The $E_{sa,j}$ and $E_{il,j}$ quantify the total surrogate model error due to the SoGP, the truncated KLE and the survivability surrogate model for fragment k

In Figure 7.3, the E_{sa} errors is represented for the six fragments that may either survive or disintegrate completely depending on the input conditions. We plot the errors in function of the number of training samples added with the active learning strategy. The additional training samples reduce the survivability assessment error by at least a factor of five up to a factor of ten. On the other hand, the impact error (not represented here) is not significantly reduced by the additional training samples. This behavior is expected since the active learner focuses on the GP classifier rather than the GP regression.

To summarize, the errors in the survivability assessment, the impact longitude, and latitude are reported in table 7.4. The computed error includes the contribution from the SoGP, the truncated KLE, and the survivability surrogate model. The errors are obtained with 2400 validation samples. For all fragments, survivability assessment error is well below 10^{-2} and close to 10^{-3}. Note that due to the limited number of validation samples and the level of misclassification error we try to estimate, the error estimate is just an indication of the degree of accuracy achieved by our surrogate model.

The impact location error depends strongly on the fragment considered. For instance, the impact location of the combustion chamber features an error level above 0.01 degree. An error of 0.01 degree corresponds approximately to an error of 1km on the Earth surface at the equator. This error should be compared to the dispersion of the fragments that are around several hundred kilometers.

Regarding the fragments for which all the training samples lead to complete disintegration, we assumed that the fragments never reached the ground. An additional 100 000 samples were generated and evaluated with the survivability solver to assess the validity of this assumption and to quantify the associated error. Out of 29 types of fragments, 17 have been found to systematically burnt up for all the 100 000 samples. The others have a probability of reaching the ground significantly below 10^{-3}. This error is below the survivability assessment error E_{sa} computed for the other fragments, and therefore it is acceptable to neglect them for this study.

7.5.2 Impact probability

For each fragment, we compute the probability that it reaches the ground. For a significant number of fragments, this probability is set to 0 since none of the training samples reached the

fragment	error longitude [deg]	error latitude [deg]	error survivability [-]
Payload adapter	3.2e-03	3.8e-04	1.2e-3
Cold Gas Reacting System	3.9e-03	4.8e-04	2.5e-3
Engine Thrust Frame 3	5.5e-03	6.7e-04	2.5e-3
Helium spheres	8.4e-03	9.5e-04	0
Combustion chamber	6.6e-02	1.2e-02	1.7e-3
Turbopump 1	1.5e-02	2.1e-03	2.9e-3
Turbopump 2	4.7e-03	5.5e-04	3.7e-3

Table 7.4: Surrogate model error for the impact conditions (longitude,latitude, survivability) after active learning. The mean square root absolute error is computed using 2400 validation samples. It corresponds to the total error induced byt the SoGP, KLE and the survivability surrogate model.

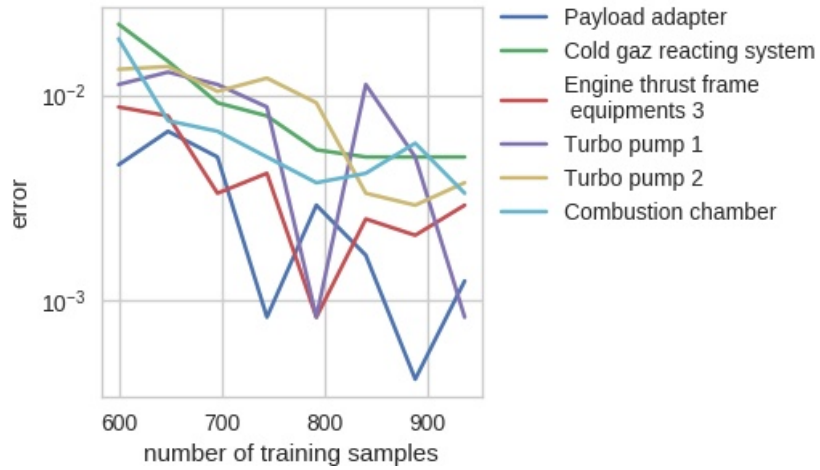


Figure 7.3: Evolution of the survivability assessment error function of the number of samples added adaptively

ground. The survivability probabilities are reported in table 7.5.

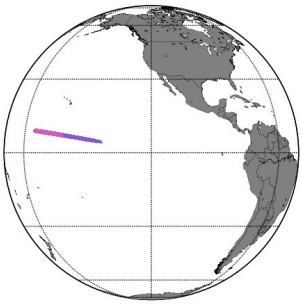
Most of the fragments that burn up completely are aluminum fragments. The aluminum low fusion temperature imposes a low radiative heat flux compared to other material. On the other hand, the pressure spheres, the nozzle and the gas chamber are most likely to survive as they are modeled with more robust materials such as titanium and inconel.

7.5.3 Impact location

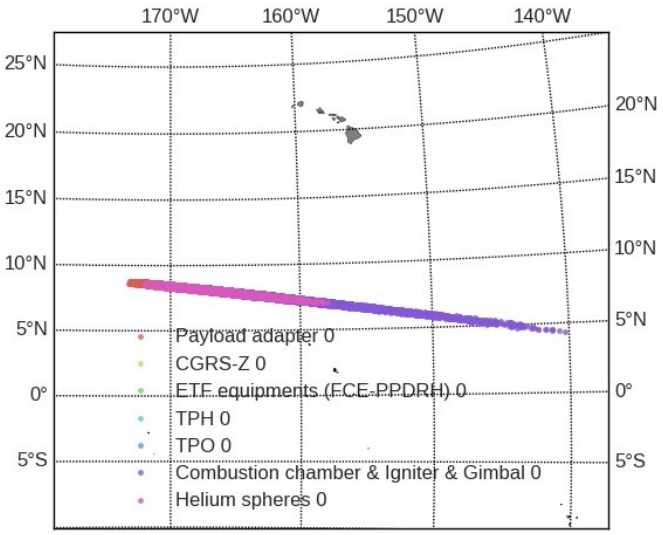
In this section, we focus on the impact distribution generated by the surviving fragments. Some fragments are identical (for instance the pressure spheres), in this case, we assume they fall in the same location. The impact locations are indexed in terms of latitudes and longitudes. We denote E the joint set of longitude and latitudes on Earth. Hence $E = [-180, 180] \times [-90, 90]$. In figures 7.4a and 7.4b, the impact locations of the surviving fragments are represented. All of them fall in the Pacific Ocean, South of the Hawaii archipelago and North of Washington and Tabuaeran Islands. The area covered by the fragment impacts is approximately 260 000 km^2. The impact distribution is elongated along one direction given by the initial orbit. The dispersion of the impact point depends on the fragment considered. The most dispersed impacts come from the nozzle that is the heaviest and largest fragment. The pressure sphere impacts are

Id	Number	Name	impact probability
1	4	Payload adapter rings	0
2	1	Payload adapter	**0.505**
3	4	Cone	0
4	4	Equipments plate	0
5	3	Equipment bay structure 1	0
6	3	Equipment bay structure 2	0
7	4	Payload adapter Ring	0
8	4	Equipment bay structure 3	0
9	4	LH2 tank upper skirt 1	0
10	4	LH2 tank upper skirt 2	0
11	8	LH2 upper dome	0
12	4	Equipment bay structure 4	0
13	3	LH2 tank cylinder	0
14	4	LH2 tank lower skirt	0
15	4	LH2 tank ring	0
16	8	LH2 tank lower dome	0
17	3	Lox tank lower dome	0
18	3	Lox tank upper dome	0
19	2	Cold Gas Reacting System	**0.12**
20	8	Engine Thrust Frame 1	0
21	1	Engine Thrust Frame 2	0
22	1	Engine Thrust Frame 3	**0.045**
23	3	LH2 feedlines	0
24	3	LOX feedlines	0
25	4	Helium spheres	**1**
26	1	Combustion chamber	**0.95**
27	3	Power unit	0
28	1	Turbopump 1	**0.44**
29	1	Turbopump 2	**0.41**

Table 7.5: Survival probability for each fragment



(a) Impact location on spherical Earth



(b) Zoom on the impact location

Figure 7.4: Impact location

also much more dispersed than the Engine Thrust Frame. In general, light fragments with low survival probabilities feature small dispersions of their impact locations.

To assess the dispersion of the fragment impacts for a given set of uncertainty values ξ, we can compute the average distance between each impact for a given reentry scenario. The average impact distance is defined a :

$$D_{impact}(\xi) = \frac{n_{sur}(n_{sur}-1)}{2} \sum_{i=1}^{n_{sur}} \sum_{j=i+1}^{n_{sur}} d_{geo}\left(x_i(\xi), y_i(\xi), x_j(\xi), y_j(\xi)\right) \tag{7.9}$$

where d_{geo} is the geodesic distance and n_{sur} is the number of surviving fragment (that depends on ξ). The quantity D_{impact} is a random variable as it depends on ξ. In Fig. 7.5, the distribution of D_{impact} is represented. The average distance between debris is most of the time between 120 km and 500 km.

7.5.4 Casualty area

The casualty area represents the potential danger for human populations if the impact location had been an inhabited area. The casualty area depends on the dimension of the object and the projected area of a standing adult. Strictly speaking, the object size reduces during the hypersonic flight due to ablation and hence the casualty area changes. Following the recommendations in [64], we compute the casualty area using the original dimensions of the fragment, which is a more conservative approach. As a consequence, the casualty area merely takes a discrete number of values depending on the surviving fragments. The distribution of the casualty areas is represented in Fig. 7.6. The average casualty area is 26 m^2. In a densely populated area (5000 hab/km^2), a casualty area of 26 m^2 leads to an expected number of victims around 0.12. At most, the casualty area can be 32 m^2. In this case, if the fragments landed over a densely populated area the calculated number of victims would be around 0.15. This maximum casualty area is associated with situations where all eleven pieces that have non-zero survival probability in table 7.5 reach the ground.

The average casualty area can be compared to the reentry of a Delta II upper stage performed in [215] where a casualty area of 18 m^2 is reported. However, the reentry conditions are very

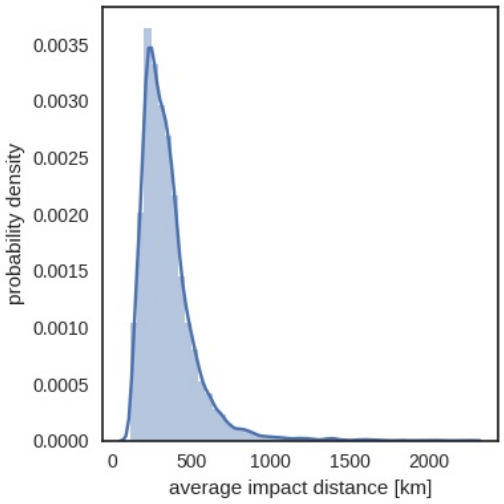


Figure 7.5: Distribution of the average distance between fragments

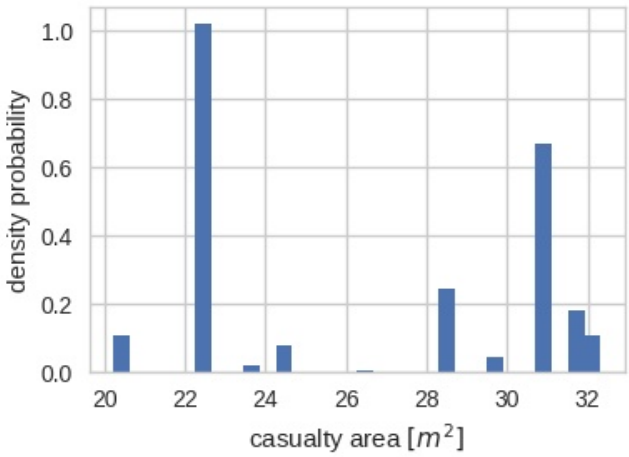


Figure 7.6: Distribution of the impact areas

different from our (LEO reentry in [215]) and the Delta II upper stage is also seven times lighter than our upper stage. In the literature, the casualty of large satellites is around 30-70 m^2 [60, 215]. The relatively small casualty area found in our case is mostly due to a large number of fragments made of aluminum that disintegrate systematically and the limited number of fragments. In particular, the propulsion unit could be modeled with a much larger number of fragments resulting in a much larger casualty area. An interesting extension of this work would be to model the uncertainties in the fragment characteristics (number, shape, material composition).

As for the breakup predictions, we are also interested in the sensitivity of the predicted casualty area to the uncertain inputs. In Fig. 7.7, the total Sobol indices for the casualty area variance are reported. The total Sobol index represents the contribution to the casualty area variance of a given input uncertainty. It includes all the contributions induced by the input variable, including the contributions generated by interactions with other variables.

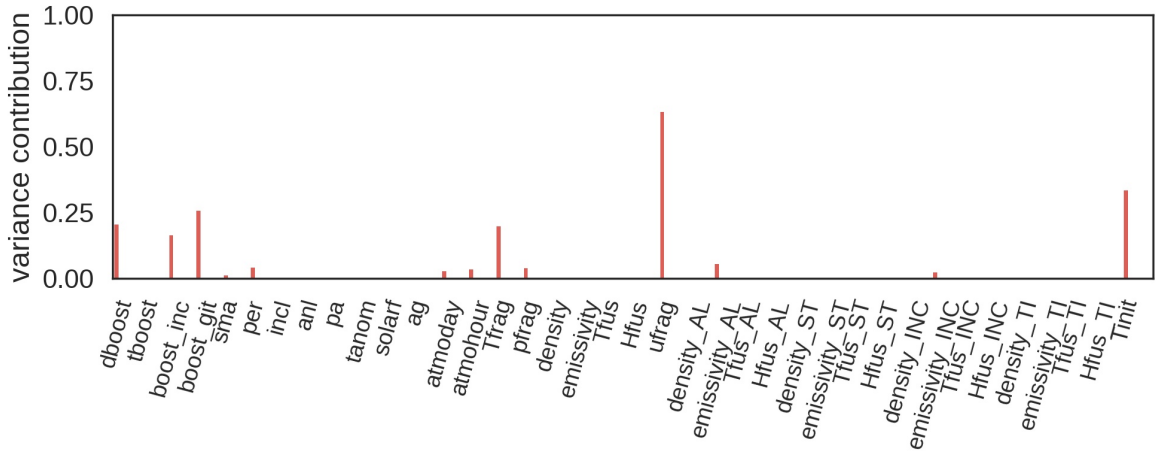


Figure 7.7: Total Sobol indices for the predicted casualty area

We observe that the main input uncertainties are related to the breakup prediction, namely the initial fragment temperature T_{init}, the quantity u_{frag},defined in 5.2 and T_{frag}. The variable u_{frag} parametrizes the probabilistic aspect of the breakup model. The variable T_{frag} characterizes the breakup time lower bound and hence influences the fragment initial position and velocity. From chapter 5, we know that the u_{frag} and T_{frag} affect the breakup flight conditions that in turn affects the fragment survivability. The quantity T_{init} is also crucial. Intuitively, it represents the amount of energy the fragment received before being released. As expected, this quantity strongly affects the survivability of the fragment as it significantly modifies the maximum amount of energy the fragment can receive during their reentry before disintegrating completely.

All those uncertainties come from the poorly calibrated and oversimplified breakup model. The variable u_{frag} represents our inability to predict the breakup within a given time interval (see section 5.4.5, Eq. (5.2) and section 3.2.2 for the detailed presentation of the breakup model). The uncertainty in T_{frag} represents our inability to characterize this time interval well, while the fragment release temperature T_{init} embodies the uncertainty induced by the simple thermal model used to compute the object thermal response. Recall that a uniform temperature model is used to calculate the thermal response of the upper stage. With this model, it is merely possible to derive a temperature interval for the released fragments.

The other set of relevant parameters are the deorbiting boost conditions. The deorbiting boost characterizes the initial reentry conditions and consequently the breakup flight conditions. High-velocity reentries with limited slope lead to high altitude fragmentation and reduced casualty area while steep reentries lead to low breakup altitudes and large casualty areas.

The third group of influential variables is the fragment material uncertainties, in particular, the aluminum and Inconel emissivity. Among all material characteristics, the emissivity is the least known quantity due to the oxidation effects that alter the material surface properties during the reentry. For this reason, the emissivity has more influence on the output than the other material characteristics. The impact of the aluminum emissivity comes from the Engine Thrust Frame 3 (ETF 3) that is a large aluminum object. Depending on the material emissivity, the fragment may be able to cool down efficiently by radiation and survive. Since it is a large object, it also contributes significantly to the casualty area. The same remark can be formulated for Inconel fragments. Note that the atmosphere model parameters also have a minor influence on the casualty area uncertainty as that they only affect the upper stage trajectory before the breakup.

7.6 Conclusion

In this chapter, we coupled several uncertainty propagation tools with the simulator developed in Chapter 3 to provide robust estimates of the on-ground risk associated with the controlled reentry of an upper stage from a GTO orbit. The simulation included the entire reentry event from the deorbiting manoeuver to the impact of the surviving fragments. A rigorous uncertainty quantification study was performed to derive robust estimates of the risk metrics as defined in the ESA guidelines and the French space operation act. In this study, we considered 38 independent uncertainties including the initial orbit elements, the deorbiting conditions, the atmosphere model parameters, breakup model parameters, and material characteristic uncertainties. Given this set of uncertainties, we derived the casualty area distribution. We performed a sensitivity analysis of the casualty area based on the Sobol indices.

We constructed a global surrogate model of the reentry simulator. The construction relied on two original surrogate modeling approaches developed in this work : the SoGP framework (Chapter 4) and the risk estimation surrogate model (Chapter 6). The deorbiting solver and the trajectory solver are approximated with a SoGP to learn the coefficients of a KLE decomposition of the trajectory. We constructed a surrogate model for each fragment to predict the survivability and impact location of each fragment. The active learning strategy developed in Chapter 6 have been extended to the case of predictions with multiple fragments to improve the predictions in the total casualty area. The surrogate model accuracy has been assessed with a validation set. The error in the survivability assessment for the surviving fragment is around 10^{-3}, and the maximum error in the impact location is around 1km.

Based on the surrogate models, we computed the impact location and survival probability of each fragment. Out of 29 types of pieces, only seven have a probability of reaching the ground significantly larger than 0 ($\geq 10^{-3}$) while the rest always disintegrate in the atmosphere. The fragments the most likely to survive are the pressures sphere, nozzle and combustion chamber in good agreement with on-ground observations. We computed the distribution of the casualty area that is an indicator of the harm the object can cause when landing in a densely populated region. Based on the sensitivity analysis, the most influential variables are the initial deorbiting conditions and the breakup model parameters.

In this work, we considered a large number of uncertainties, but not all uncertainties were included. For instance, the uncertainties in the aerodynamic models for the fragments or the influence of winds have not been included for example. Moreover, the casualty area analysis suggested that the number of fragments and their properties should be selected as uncertain.

The uncertainty quantification results of this chapter rely on surrogate models with minimal L_2 error. However, they are not well suited for computing rare events. The surrogate modeling approaches could be adapted to compute low probability events accurately using importance sampling and advanced active learning techniques [168, 211, 212, 213, 214].

Chapter 8

Conclusions and Perspectives

Space activities provide critical services for human economic activities such as communication broadcasting or Earth observation. This growing interest in space led to a steady increase in the number of in-orbit space object to the point where future space activities become compromised by the in-orbit collision risk with space debris. International guidelines recommend that end-of-life satellites reenter the Earth atmosphere within 25 years with a controlled on-ground risk. Predicting the risk is extremely challenging as it relies on approximate models. Some phenomena, such as for example fragmentation, cannot be described with trustworthy models. The predictions performed with those models are unreliable due to the necessary approximations and assumptions that alleviate the computational burden, but at the price of a bias in the result.

8.1 Conclusions

In this work, we proposed several contributions for improving the computation of the risk with a space object reentry simulator. We worked on three main aspects: i) the construction of a system of solvers using the reentry solvers of AG; ii) the development of several uncertainty quantification methods tailored to the propagation of uncertainties through generic directed systems of solvers and the computation of the on-ground risk distribution; iii) the application of these methods to a realistic debris reentry scenario. By coupling the different UQ methods with the reentry simulator, we were able to provide a prediction under uncertainty of the reentry of an upper stage and to derive at a reasonable computational cost the on-ground risk statistics and identify the most critical uncertainties using global sensitivity analysis.

We demonstrated the performance of the UQ techniques we developed on an industrial problem. We illustrated the capability of our reentry tool to robustly compute the on-ground risk associated with a space object reentry.

The main contributions are summarized here:

- We developed a reentry simulator based on several solvers from AG. We managed the model complexity and multi-physics aspects by constructing a system of solvers. The simulator is implemented in a flexible, user-friendly, parallel framework that automatically handles interactions between solvers and facilitates the application of uncertainty quantification strategies.
- We enhanced the predictive ability of the simulator with a probabilistic breakup model that naturally handles the uncertainties induced by the extreme complexity of the object

breakup.

Object-oriented models do not model the breakup and use a fixed breakup altitude. On the other hand, spacecraft-oriented models couple finite element models, hypersonic flow models and engineer expertise to predict the breakup. This higher level of accuracy comes with a significant computational cost and user interaction that prohibits the use of uncertainty propagation methods. For this reason, spacecraft-oriented models are usually used in a deterministic setting with limited confidence in the result.

In this work, we propose a third, intermediate approach based on probabilistic modeling of the breakup. We believe that an accurate breakup model is still hard to formulate and therefore breakup models should not be deterministic. We applied this idea in this work with a straightforward model of the breakup. We also developed a more complex model to provide more realistic breakup predictions but still in a probabilistic setting (see Appendix B).

- We proposed an efficient surrogate model strategy for directed systems of solvers. Systems of solvers are widely used for industrial multi-physics problems but are complicated to approximate due to their high input dimensions. We proposed to approximate a system of solvers with a system of Gaussian processes where a GP approximates each solver of the SoS. For all the numerical tests performed on analytical functions and real industrial applications, our system of Gaussian process outperforms the single GP (black box) approach systematically. Our method is more efficient than a black box approximation of the SoS because it builds several surrogate models (one for each solver) that are usually easier to learn that the entire system, especially if the input dimension of each solver is smaller than the overall SoS input dimension.

  Additionally, we showed that the SoGP could naturally generate parsimonious training sets. We proposed a decomposition of the predictive variance of the SoGP into contributions of each GP. Based on this decomposition, we proposed active learning strategies that can enrich the training set of specific GPs based on their contributions to the SoGP predictive variance. Numerical experiments clearly showed the advantage of using such active learning strategies in particular for medium input dimension problems.

- We proposed an original formulation of the mapping between the uncertain inputs and the estimated risk viewed as the joint prediction of a classifier and a continuous function. This formulation led to the construction of a surrogate model using a GP classifier and a GP regression. It permits to handle the survivability assessment and impact location prediction in a unified way whereas state-of-the-art solutions only proposed surrogate models for the impact location prediction. Additionally, active learning strategies based on the predicted on-ground risk have been introduced to improve the surrogate model accuracy. The proposed enrichment criterion naturally balances the error contributions from the GP classifier and the GP regression. The numerical tests showed the significant improvements observed when using the active learning strategies compared to randomly selected training samples.

- We performed a rigorous uncertainty analysis of a reentering upper stage combining the reentry simulator and the surrogate modeling tools developed in this work. We illustrated the advantages of our SoGP framework and survivability surrogate model on a full-scale multi-physics industrial problem by performing a on-ground risk prediction under uncertainty with only a few hundred solver evaluations (*i.e.*, a few hours of computation on a single node).

While performing this analysis, we were particularly careful in identifying and modeling the uncertainties associated with the reentry of a space object.

The significant model errors we identified are the simplified analytical formulae to compute the aerothermal loads especially and the breakup modeling. Experimental observations suggest that the object breakup is a sequence of fragment release triggered by the combined effects of aerodynamic efforts, thermal loads depending on the object trajectory and attitude motion. On the other hand, object-oriented models generally feature a single breakup event triggered at a fixed altitude independently of the object trajectory and motion. Even spacecraft oriented software can only afford breakup model based on thermal criteria.

The second type of uncertainties is scenario uncertainties. Those uncertainties are associated with the scenario at hand. They are the unknown initial conditions, the object characteristics (mass, shape mass distribution, material composition), the turbulent atmosphere properties and the material characteristics (thermal and structural properties).

While we could not include all those uncertainties in our uncertainty propagation, we considered an unprecedented number of uncertainties (38) including the initial orbit elements, the deorbiting boost characteristics, the atmosphere model parameters, the upper stage material characteristics, the breakup model parameters including the fragment release temperature, and the fragment material characteristics. We studied, in particular, the evolution of the uncertainties along the trajectory before the breakup, the object flight conditions at breakup, the casualty area, and the fragment impact locations. We also performed a rigorous sensitivity analysis using Sobol indices. We observed that the breakup flight condition uncertainties were mostly due to the breakup model uncertainties and the deorbiting conditions while the atmosphere model uncertainties were secondary. The sensitivity analysis of the casualty area also identified the deorbiting conditions and the breakup model parameters as influential variables. This analysis suggests that the breakup model should be improved or the deorbiting conditions better characterized if more precise predictions were necessary.

The framework developed can readily be used for demonstrating the compliance of the deorbiting manoeuver with respect to the French Space Operation Act with increased robustness compared to existing deterministic reentry tools.

8.2 Perspectives

Basing on the work done, we propose here several research lines. The first one is based on the system of Gaussian process framework while the others are oriented toward the reentry prediction problem.

8.2.1 Developments of the SoGP framework

One essential contribution of this work is the introduction of a system of Gaussian processes for emulating system of solvers. While showing promising results, our implementation of the SoGP could be improved.

The active learning strategies based on the variance decomposition of the SoGP predictive distribution showed promising results. Nevertheless, the active learning criteria based on MMSPE showed weaknesses due to the MMSPE criterion itself. Further work could work on implementing more efficient models such as the IMSPE criterion or the mutual information criteria. Computationally efficient methods proposed in [183] for computing the IMSPE criterion could be adapted to the SoGP framework using the variance decomposition.

SoGPs can currently only approximate directed system of solvers without strong coupling between solvers. A significant contribution would be to extend the SoGP framework to a strongly coupled system of solvers.

The SoGP framework does not have to be used solely for uncertainty propagation purposes. An exciting development would be to adapt it to optimization problems. Bayesian optimization is an efficient approach derivative-free method for solving non-convex optimization problems that relies on a probabilistic approximation of the functional to optimize to efficiently find the optimum [216]. The SoGP framework seems well indicated for performing Bayesian optimization in a system of solvers. The active learning criteria (called acquisition functions) such as expected improvement [167], entropy search [217] etc. could be adapted to SoGPs and their variance decomposition.

The second use of SoGP is for reliability engineering. For our application, we would be interested in computing low probability events, but the constructed SoGP is not well suited for such problems as it provides a global approximation of the reentry simulator. Future work could focus on developing dedicated methods based on active learning to construct a SoGP trained specifically to compute a given low probability event. For single GPs, many approaches have been developed, but they need to be extended to SoGPs.

The surrogate model constructed in Chapter 7 to emulate the entire reentry simulator is not a SoGP, hence we could not apply the results derived from Chapter 4 as for instance the variance decomposition. One interesting development would be to extend this framework to a larger class of surrogate models than GPs. It seems that as long as the solvers are approximated by independent probabilistic surrogate models, the variance decomposition holds (and hence the active learning strategies we developed), even is the surrogate model predictive distribution is not Gaussian.

8.2.2 Improving the reentry predictions under uncertainties

We propose to improve the reentry simulator by refining the physical and statistical models of the breakup solver and improving the uncertainty modeling.

We proposed to model breakup as a sequence of random events. We believe that this approach has a high potential of offering reliable breakup models using more advanced physical models and statistical modeling. A possible approach would be to model breakup as a probabilistic fault tree [218] that spans a distribution of breakup scenarii, including sequential fragment releases. In [61] a Bayesian failure tree is proposed, but it relies solely on a statistical model and missed the physical models and uncertainty propagation tools developed in this work. Using the framework developed in this work, one could build a probabilistic fault tree based on physical models and if available, experimental data. This approach should aim not only at deriving the fragment release distributions but also at considering uncertain fragment shapes and number. We expect probabilistic breakup models to be more demanding in terms of computer resources than deterministic ones. Hence we believe that a probabilistic breakup solver should be coupled with efficient surrogate modeling approaches as we did in this work.

Some high fidelity models used in spacecraft oriented tools as the 6 DoF trajectory solver could not be approximated with a GP due to the chaotic attitude motion of the upper stage during the reentry. The object attitude motion and subsequently its trajectory are extremely sensitive to even small variations in the initial conditions and aerodynamic moments. Constructing a surrogate model is hence extremely challenging. One approach could be to try other families of surrogate models such deep neural networks [219] or even a solver-free resolution of the trajectory [220, 221, 118]. Alternatively, a multi-fidelity approach could be used by formulating the 6 DoF trajectory as the 3 DoF trajectory plus a random perturbation to that needs to be characterized

(time and space correlation, amplitude).

In Chapter 2 we acknowledged the importance of model errors in the aerodynamic and thermal models, but they were not included in the uncertainty analyses we performed. One approach would be to model the model errors with an additive Gaussian Process with a given kernel and set of hyperparameters. The error level is *a priori* not constant (the aerodynamic calculations tend to be more accurate in the free molecular flow regime than in the rarefied regime). Hence the GP would be non-stationary. Once the error modeled are calibrated, they could be included in the system of solvers and the system of Gaussian processes. While the construction of the SoGP should remain unchanged, the strategies for generating efficient training sets would undoubtedly need to be revisited.

8.2.3 Detecting low survival probability fragments

Some fragments have very low survival probability, and training a classifier to predict survivability in this case is extremely challenging. This problem is known as anomaly detection in the machine learning community [144]. Sometimes, even being able to generate a training set with samples from both classes is hard. Random sampling approaches can be extremely inefficient if the survival probability is below 10^{-3}. One approach to detecting survival efficiently would be to build a SoGP predicting the demise altitude function of the random inputs and adding a training sample where the predicted demise altitude is minimal. The intuition is that the fragment has better chances of surviving where the predicted demise altitude is low. Preliminary results showed promising results where sporadic survival cases could be detected using this simple physical criterion.

8.2.4 Computing impact zones for multiple fragments

In chapter 2, we defined the impact zone as the smallest set in terms of the area containing a given probability weight. In [66, 222], an elegant solution solution to this problem based on minimum volume sets is proposed. However, the method described in [66] can only be used for a single fragment case whereas in most reentries feature several fragments. We propose to directly find the impact zone A as the solution of an area minimization problem with a constraint on the probability weight contained in A. One could use shape optimization strategies [223] and use efficient reliability techniques applied to the reentry simulator and our surrogate model to efficiently compute the constraint.

8.2.5 Design for demise

Design for demise strategies aims at minimizing the on-ground risk by optimizing the object characteristics (material, aerodynamic shape) and if possible the flight conditions. It can be formulated as an optimization problem where the deorbiting conditions and the object characteristics are optimized to minimize the on-ground risk. For now, design for demise usually focuses on local solutions (demisable joints, epoxy glue) without a general view of the full reentry problem. Based on this work, the reentry simulator and the surrogate model strategies developed in this work could be reused in an optimization framework to find the upper stage characteristics and deorbiting conditions that minimize the on-ground risk. By constructing a complete feedback loop with our model, the design for demise solution could be rapidly numerically validated and optimized.

Appendices

Appendix A

ArianeGroup solvers

In this section, we present the tools developed by ArianeGroup and the CNES and selected to construct the reentry simulator presented in Chapter 3. In total four different solvers are used.

A.1 Deorbiting solver : CARINE

Most reentry models start with the object flight conditions at 120 km altitude. In this work, we want to predict the reentry of an object from its initial orbit if it is deorbited in a controled way. The deorbiting solver computes the object trajectory between the initial orbit and 120 km of altitude. We use solver CARINE that is an orbit propagator that computes trajectories after a deorbiting boost is released. It takes as input the orbital elements of the initial orbit and the deorbiting characteristics. Namely the deorbiting time t_{boost}, the retro boost amplitude A_{boost} and the boost orientation orientation characterized by two angles, *i.e.* α_{boost} and β_{boost}. The new orbit characteristics are evaluated, and the orbit propagated until the object reaches 120 km of altitude. The model is less expensive than state-of-the-art orbit propagator such as STELA from CNES [70] but it is only valid for short time propagation as for a deorbiting trajectory.

A.2 Aerodynamics/Aerothermal solve : ARPEGE

To solve for the aerothermal load applied to the spacecraft, the software ARPEGE has been developed by ArianeGroup and CNES. It computes the aerodynamic forces, moments and convective heat flux for a given object geometry, free stream conditions, object velocity and attitude motion. For the aerodynamic forces and moments, the approach is similar spacecraft-oriented models presented in chapter 3 where $\boldsymbol{F_{aero}}$ and $\boldsymbol{M_{aero}}$ are defined as :

$$\boldsymbol{F_{aero}} = \int_S (\boldsymbol{p} + \boldsymbol{\tau}) dS \tag{A.1}$$

$$\boldsymbol{M_{aero}} = \int_S (\boldsymbol{r} \times \boldsymbol{p} + \boldsymbol{r} \times \boldsymbol{\tau}) dS \tag{A.2}$$

where $\boldsymbol{p}$ is the local pressure, normal to the surface and $\boldsymbol{\tau}$ is the shear stress, tangent to the surface. The solver uses a local panel method: the object is represented by a set of elementary surface panels for which $\boldsymbol{p}$ and $\boldsymbol{\tau}$ are computed. The forces are obtained by integrating the contributions from each panel.

The formulation of pressure and shear stress depends on the flow regime. Three flow regimes are observed during a reentry: the free molecular flow, the rarefied flow, and the continuum flow. The free molecular flow exists at high altitudes and low densities, where the molecular mean

free path is comparable to the object size. In this regime, analytical expressions of the pressure and shear stress hold using [55] (eq. 7.88-7.89) and injected in Eq. (A.1). In the continuum hypersonic flow regime, the modified Newton law holds [4] for the pressure forces while the shear stress is set to zero. The rarefied flow is the transition between the free molecular flow and the continuum flow is computed using bridging functions designed by ArianeGroup.

ARPEGE returns the aerodynamic coefficients for forces and moments that can be collected to build tables function of the Mach number , the Knudsen number and the object orientation. In 3 degree of freedom setting, the aerodynamic coefficients are the drag coefficient C_D, the lift coefficient C_L, the side force coefficient C_Y. The relationship between the aerodynamic coefficients and the corresponding aerodynamic force is given by:

$$F_* = \frac{1}{2}\rho_\infty V_\infty^2 C_* S_{ref} \tag{A.3}$$

with ρ_∞ and V_∞ the free stream density and velocity, and S_{ref} the object reference area chosen by the user. The sufix $*$ denotes the indices of the drag,lift or side force coefficient. Note that ARPEGE can also be used in a 6 DoF setting and compute the aerodynamic moments. In the appendix, we propose a high fidelity reentry simulator with a 6 DoF resolution of the trajectory.

The solver ARPEGE is also able to reconstruct the flow characteristics in the shock layer (enthalpy, pressure, viscosity) and to derive the convective heat flux from the fluid to the object.

A.3 Trajectory solver

The trajectory solver used in this work is BL43 developed by AG. It is a trajectory module for reentry trajectories that we use between 120 km altitude to sea level. It can be used in 3 DoF or 6 DoF. In that case BL43 solves for :

$$\frac{d^2\boldsymbol{X}}{dt^2} = A_e + A_c + A_g + \frac{\boldsymbol{F_{aero}}}{m} \tag{A.4}$$

$$I \cdot \frac{d\boldsymbol{\Omega}}{dt} = \boldsymbol{M_{aero}} - \boldsymbol{\Omega} \wedge (I \cdot \boldsymbol{\Omega}) \tag{A.5}$$

The quantity F_{aero} is reconstructed using aerodynamic coefficient tables provided by the user and equation (A.3). The free stream density and velocity are computed using the MSIS00 atmosphere model. The free stream conditions computed by the atmosphere model depend on the object position, the reentry time, solar activity and Earth geomagnetic index that have to be provided. An option can also be activated to account for the wind effects in the subsonic part of the trajectory. Given the initial position and velocity at 120 km and the atmosphere model parameters, BL43 returns the trajectory of the object denoted by its position and velocity vector $\boldsymbol{X(t)}$, $\boldsymbol{V(t)}$ in a 3 DoF setting and position, velocity Euler angles and derivatives ($\boldsymbol{X(t)}$,$\boldsymbol{V(t)}$,$\boldsymbol{\Theta(t)}$,$\boldsymbol{\dot{\Theta}(t)}$) in a 6 DoF setting.

A.4 Thermal solver : ADRYANS V4

ADRYANS V4 [206] is a thermal model used along a trajectory to compute the heat flux q_{aero}. It calculates the aerothermal heatflux and the temperature evolution along the trajectory. It is also composed of an ablation module that compute the mass evolution along the trajectory. It is developed by AG and CNES. The model assumes uniform object temperature and neglects metal oxidation or molecular recombination at the surface of the object. The uniform temperature

assumption has been widely used in object-orientated software [75]. Hence, energy conservation equation for the entire object yields:

$$m_{ob}c_p\frac{dT_{ob}}{dt} = Sq_{aero} + \delta P_{out} \tag{A.6}$$

where T_{ob} is the object temperature, m_{ob} its mass,S its surface,c_p the heat capacity, q_{aero} the aerothermal heatflux and δP_{out} is the power loss due to ablation. In the rest of the paragraph, we detail the calculation of q_{aero} and δP_{out}.

The aerothermal heat flux q_{aero} is the sum of two contributions :

$$q_{aero} = q_{conv} + q_{rad}, \tag{A.7}$$

where q_{conv} is the convective heat flux and q_{rad} is the radiated heat flux. Compared to spacecraft oriented models, the conductive heat flux and the oxidation heat flux are not modelled

To recover the averaged convective heat flux over the object, we simplify its geometry. For upper stages, we use a cylinder shape, for satellites, a box shape. Then ADRYANS uses correlation formulae close to [77] to relate the convective heat flux received by the object with the heat flux at the stagnation point of a flat plate for which analytical formulations exists. For instance, in the free molecular flow regime, the convective heat flux at the stagnation point of a perpendicular flat plate is [77]:

$$\dot{q}_{sp}^{FM} = \frac{\alpha\rho_\infty V_\infty^3}{2} \tag{A.8}$$

where α is the accommodation coefficient, ρ_∞ and V_∞ are the free stream flow density and velocity.

In continuous regime, at the stagnation point of a perpendicular flat plate, the Detra-Hidalgo correlation gives [78]:

$$\dot{q}_{sp}^{cont} = \frac{C}{\sqrt{R_N}}\left(\frac{V_\infty}{V_c}\right)^{3.15}\left(\frac{\rho_\infty}{\rho_s}\right)^{0.5} \tag{A.9}$$

with ρ_s being the atmosphere density at sea level,V_c the circular orbit velocity and R_N the object nose radius. C is a numerical constant ($11028 \times 10^4 kg.s^{-3}$)

For the transitional regime the Matting bridging function is used [224]:

$$\dot{q}^{rar} = \dot{q}^{cont}\left(1 - \exp\left(-\frac{\dot{q}^{FM}}{\dot{q}^{cont}}\right)\right). \tag{A.10}$$

The radiative heat flux is computed using Stefan-Boltzmann law:

$$\dot{q}_{rad} = \sigma\epsilon(T^4 - T_{env}^4) \tag{A.11}$$

where σ is the Stefan Boltzmann constant, ϵ is the material emissivity, T is the temperature and T_{env} the environment temperature set to 4K. Note that the radiative gas effects in the shock layer are neglected.

The ablation model implemented in ADRYANS V4 is similar to the object-oriented models. When the T_{ob} reaches fusion temperature, the energy brought by the flow melts a parts of the object that are swept by the flow. Hence, we have

$$P_{out} = \dot{m}_{ob}h_{fus} \tag{A.12}$$

where $\dot{m}_{ob}$ is the mass loss rate and h_{fus} is material fusion enthalpy. Solving for m_{ob} and T_{ob} in (A.6), ADRYANS returns the object temperature and mass over time. Note that this approach is only valid for metal alloys.

Appendix B

High fidelity reentry simulator

In this section we present the high fidelity reentry predictor developed in this work. The reentry predictor is based on recent developments of ArianeGroup models, namely ADRYANS V5.0.

B.1 General description

The model presented here was developed to remove some of the assumptions employed in the reentry simulator developed in Chapter 3 that hereafter is referred as the medium fidelity simulator. It is based on improvements in the models developed at ArianeGroup within particular ADRYANS V5.0 and new features in ARPEGE to perform local the convective heat flux calculations for arbitrary shape. The global predictor architecture shares similarities with the medium fidelity model. It depends on whether the reentry is controlled or uncontrolled. The model architectures are detailed in figure B.1. The major differences with the medium fidelity model are the trajectory solver and the breakup simulator that are described in detail in the following sections. If the reentry is controlled, then the orbit propagator model is called (see fig B.1). Otherwise, the simulation starts at 120 km of altitude. The sequence of solver runs can be summarized as follow

1. If the reentry is controlled, the initial orbit and the deorbiting boost characteristics are given to the deorbiting tool that computes the object flight conditions at 120 km.
2. The aerodynamic coefficients associated with the object geometry are computed by the aerodynamic solver (ARPEGE). In order to avoid calling the aerodynamic solver at every time step of the trajectory, tables are generated. The tables are a function of the object attitude (orientation), Mach number and Knudsen number.
3. The 6 DoF trajectory solver BL43 is used to compute the object trajectory.
4. The object position, velocity and attitude motion are used by ARPEGE to compute the convective heat flux along the trajectory
5. The object position, velocity, attitude motion, and convective heat flux are given to ADRYANS that computes the thermal response of the object
6. The breakup model uses the thermal response, the object position, and velocity to compute the fragment release times and conditions.
7. Using the precomputed list of fragments and the breakup conditions, the survivability solver (BL43 coupled with ADRYANS V4) assesses the survivability of the object and the impact location of the surviving fragments

In this model, the trajectory solves for the position and velocity but also for the attitude motion in a 6 DoF setting. The trajectory solver is not coupled with ADRYANS in this module. Instead, the trajectory $((X(t),V(t),\Theta(t),\dot{\Theta}(t))$ containing the position, velocity and the angular orientation and angular velocity is used by the solver ARPEGE to compute the convective heat flux for every cell of the mesh at every point the trajectory. Then, the thermal response from ADRYANS V5 is performed using the convective heat flux calculated by ARPEGE. The thermal response including the temperature profile, the ablation rate, density profile is transmitted to the breakup solver in order to compute the breakup time. As for the medium fidelity model, the breakup solver is probabilistic to account for the inevitable uncertainties in the breakup prediction.

In the following, we provide a detailed description of the solvers used in this high fidelity simulator.

B.1.1 High fidelity thermal module (ADRYANS 5.0)

In 2017, an improved version of the thermal model of ADRYANS was developed. The main improvements compared to 4.0, are the modelling of composite material and the modelling of oxidation and conductive heat flux with a 1D model [206].

Material modelling The thermal module can account for complex material assembly with multiple material layers connected with thin glue layers. Complex structures such as honeycomb constructions can also be considered using an extensive material database based on experimental campaigns. Moreover, advanced emissivity evolution models based on ground experiments are included. They allow accounting for the emissivity changes due to the alteration of the material from surface oxidation (see Chapter 2).

Heat flux modelling The total heat flux at the surface of the material can be decomposed into

$$q_{tot} = q_{conv} + q_{rad,in} - q_{rad,out} + q_{oxi} - q_{cond} \tag{B.1}$$

with

- q_{conv} the convective heat flux ;
- $q_{rad,in}$ the incoming radiative heat flux ;
- $q_{rad,out}$ the heat flux radiated by the object ;
- q_{oxi} the heat flux due to exothermic oxidation reaction occurring at the surface of the object ;
- q_{cond} the conductive heat flux.

Ablation modelling As in SCARAB, ADRYANS V5.0 proposes two distinct ablation models for alloys and composite materials. For the alloys, the peeling strategy used in most software (DEBRISK, SCARAB, FAST etc.) is applied. For composite materials a dedicated model is proposed. The composite material model accounts for the material swelling, the pyrolysis cooling reactions, blocking effects and material erosion.

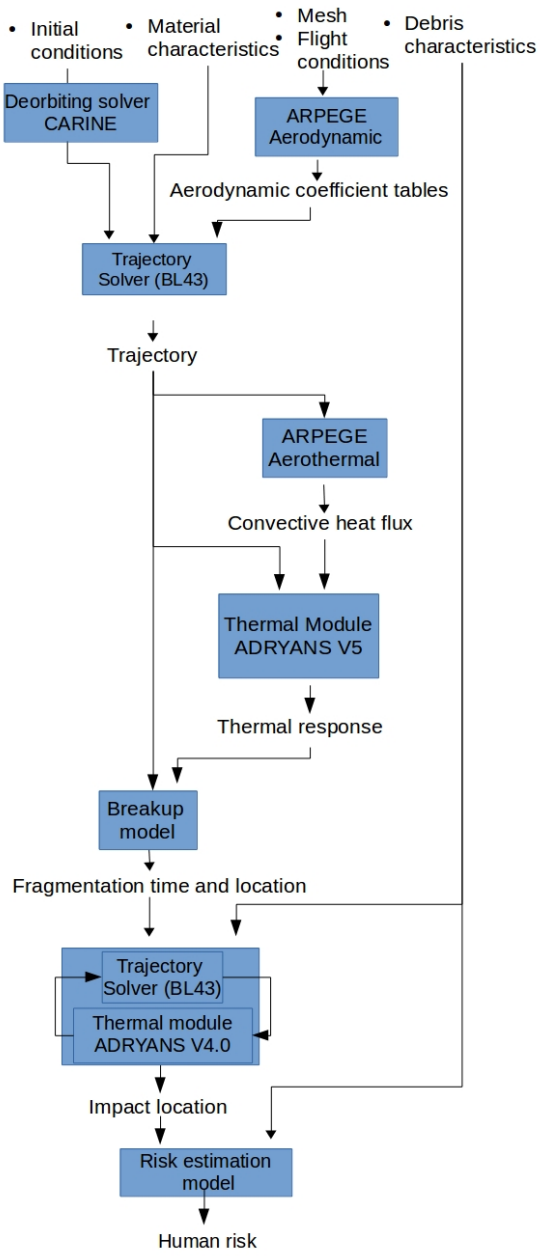


Figure B.1: Functional representation of the high fidelity simulator

B.1.2 Object geometry model

In this model, the object complex geometry is taken into account when computing aerothermodynamic load. The aerodynamic coefficients and the convective heat flux are calculated by ARPEGE based on the mesh representation of the object. Concerning the thermal response of the object, the object geometry is simplified into zones that represent distinct parts of the object. For each zone of the model, a 1D thermal response of the object is computed. The heat exchanges between the zones are neglected. The geometry and size of the zone are selected by

the user and depends on the material composition, object geometry and the regions likely to break up. In the next section, we propose a possible zone division of an upper stage into 6 zones.

B.1.3 Break up model

As for the medium fidelity simulator, the breakup model is stochastic but with a more realistic modeling of the breakup based on the computed thermal response. For each zone, we define a zone failure time $\mathcal{T}_f^i$ at which the object zone fails. A failure event includes welding failure, structure buckling, joint melting, glue melting or object ablation. When the zone fails, the fragments associated with this zone are released. As the thermal model is not sufficiently refined to model the different type of possible failure mentioned as for instance the joint melting; or it lacks a mechanical model to account for structure buckling, a probabilistic approach is selected. To account for these simplifications, the random zone failure time $\mathcal{T}_f^i$ is defined as :

$$\mathcal{T}_f^i = \mathcal{U}(t_{init}^i, t_{end}^i) \tag{B.2}$$

where t_{init}^i and t_{end}^i depend on the object trajectory and thermal response of the zone. They define the time interval when the zone fails. The thermal response from ADRYANS V5.0 is used to define at time interval where the zone can fail rather than trying to predict the exact time and type of failure that may occur. This approach greatly simplifies the problem while maintaining a certain robustness. The uniform distribution is selected as no further information is available a priori. Depending on the trajectory and the attitude motion t_{init}^i and t_{end}^i can account for the separation of two layers, the melting of a joint or the erosion of a composite that can trigger breakup. The quantities t_{init}^i and t_{end}^i also depend on the geometry of the zone and its initial mechanical constraints. They can be defined with a temperature threshold or an ablation percentage. Using a user-defined list of fragments, the fragments associated with each of the failed zone are released at the flight conditions $X(\mathcal{T}_f^i), V(\mathcal{T}_f^i)$ and temperature $T(\mathcal{T}_f^i)$. Compared to the medium fidelity approach, this model can model several breakup events and accounts for more local phenomena depending on the mesh resolution.

As for the medium fidelity model, the distribution of $\mathcal{T}_f^i$ can be modified to account for additional a priori information or to accommodate a sequential release of the fragments. As in the medium fidelity model, the t_{init}^i and t_{end}^i are uncalibrated model parameters that should be considered as uncertain.

B.1.4 A priori limitations of the high fidelity

Despite significant improvements in the models, the high fidelity simulator still makes strong assumptions :

- The aerodynamic efforts are computed using the modified Newton law in the continuous and bridging functions in the rarefied flow
- The convective heat flux is also computed using correlations at each local triangle and simplified gas state equations.
- The thermal response of the object remains 1D in each zone and hence does not capture 2D and 3D effects.
- The breakup model does not predict the shape of the fragments and does not feature any mechanical breakup modeling.

- The object trajectory is not coupled with the breakup model. When a zone fails and the fragments associated are released, the object properties (mass, aerodynamic, inertia matrix and center of mass position) are altered with consequences on the rest of the trajectory. In this model, the changes of the object are neglected assuming that the time interval between the first and the last breakup event is short (a few seconds in practice) so that the changes in the trajectory and heat flux are small.

B.2 Simulation with the high fidelity model

The test case used for the medium fidelity model is adapted to the high fidelity one. We consider the controlled reentry of an upper stage from a GTO orbit. The mesh is slightly refined as the pressure spheres are added at the rear (see fig B.2). The object surface is divided into zones where the convective heat flux is computed with ARPEGE and then the thermal response of the zone is computed using ADRYANS. The zones are represented in different colors in fig B.2. Zone 1 corresponds to the front disk, zone 2 corresponds to the truncated cone behind. Zone 3 represents the main cylindrical body. Zone 4 corresponds to the rear planar disk. Zone 5 represents the pressure spheres and zone 6 is the nozzle. Zone 1 is made of composite carbon, zone 2 is composed of a layer of composite carbon and a layer of aluminium. Zone 3 and 4 are pure aluminium. The pressure spheres are made of titanium and carbon composite. The nozzle is modelled with carbon composite.

In order to use the breakup model, we need to define failure time distribution to each zone and to assign one or several zones to each fragments. Recall that the fragments are released as soon as one of the associated failure zone fails. As in the medium fidelity model, there are 29 types of fragments presented in table B.1. The last column indicates the zone failure that provokes the fragment release. For instance the first fragment 1 will be released as soon as zone 1, 2 or 3 has failed. The choice of the zone is based on the primary location of the fragment before release. Note that the main body breakup (zone 3) provokes the release of all the fragments. Once all the fragments have been released, no further breakups are expected. Out of the 6 zones, only two are made of aluminum while the rest is made of titanium and composite carbon that are much more resistant. Zones 5 and 6 corresponds to the pressure spheres and the nozzle that are extremely resistant. For this reason, we assume that they cannot fail. For zone 1 made of composite carbon, we assume the breakup can occur when the zone erosion starts. This means the char zone becomes significantly large and subject to breakup. We have t^1_{init} : beginning of zone 1 erosion. For t^1_{end} we take the time where a percentage p^1_{abl} of the composite material is eroded. The similar criteria are selected for zone 2, where t^2_{init} is defined as the beginning of the erosion of the carbon composite layer after the aluminum layer has been ablated. The quantity t^2_{init} is defined as the time necessary to obtain a percentage p^2_{abl} of the composite material eroded. For zone 3 (the main body), breakup can occur after the temperature reaches a user fixed value $T^3_{frag,init}$ and before it reaches $T^3_{frag,end}$. Hence we have for t^3_{init} and t^3_{end}

$$T(t^3_{init}) = T^3_{frag,init} \tag{B.3}$$

and

$$T(t^3_{end}) = T^3_{frag,end} \tag{B.4}$$

For zone 4, the failure can occur when the zone internal temperature reaches $T^4_{frag,init}$ and it ends when the zone is ablated at a percentage p^4_{abl}.

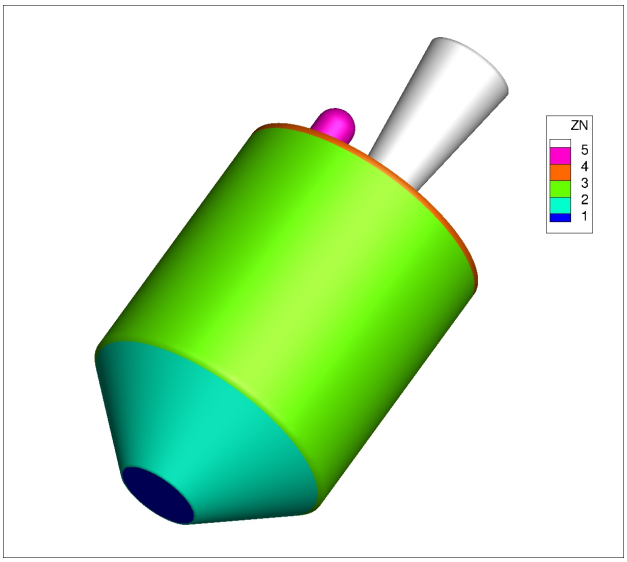


Figure B.2: Upper stage used in the high fidelity model decomposed into zones : zone 1 in dark blue, zone 2 in teal, zone 3 in green, zone 4 in orange, zone 5 in purple and zone 6 in white.

In this test, we selected $p^1_{abl} = 0.5$,$p^2_{abl} = 0.7$, $T^3_{frag,init} = 500K$, $T^3_{frag,end} = 700K$,$T^4_{frag,init} = 500K$ and $p^4_{abl} = 0.9$. As for the medium fidelity model, the zone failures occur at $\mathbb{E}\left[\mathcal{T}^i\right] = \frac{t^i_{end}+t^i_{init}}{2}$.

As for the medium fidelity model, the system of solvers predicts a reentry at 9800 m/s at 120 km altitude. The first fragmentation occurs at 68 seconds due to a failure in zone 4 (back of the upper stage) and a second breakup at 3 seconds after. During the first breakup at 64 km, the propulsion system fragments are released with among them the pressure spheres, the nozzle and the turbo pumps. During the second breakup at 65 km altitude and Mach 30, the rest of the fragments are released as the main body (zone 3) has failed. The velocity and inclination of the original object are represented in fig B.3 and B.4 . Out of the 100 fragments only the 4 pressure spheres, the nozzle with the combustion chamber, a payload adaptor and the turbo pumps survive. The fragments fall in the Pacific ocean with significant impact differences with the medium fidelity simulator. In fig B.5, the impact location of the fragments are represented. In particular the payload adapter, released during the second breakup event falls at 170 deg West instead of 168 deg West in the medium fidelity model. Note that due to the separate breakup events, the impact points are more scattered than for the medium fidelity model.

This primary test does not fully illustrate the capabilities of the simulator that is inherently probabilistic and hence a large number of breakup events should be sampled to get a meaningful representation of the impact location. This test is only preliminary and more realistic testing should be performed.

Id	Number	Name	Zone
1	4	Payload adapter rings	1,2,3
2	1	Payload adapter	1,2,3
3	4	Cone	1,2,3
4	4	Equipments plate	1,2,3
5	3	Equipment bay structure 1	1,2,3
6	3	Equipment bay structure 2	1,2,3
7	4	Payload adapter Ring	1,2,3
8	4	Equipment bay structure 3	1,2,3
9	4	LH2 tank upper skirt 1	2,3
10	4	LH2 tank upper skirt 2	2,3
11	8	LH2 upper dome	2,3
12	4	Equipment bay structure 4	1,2,3
13	3	LH2 tank cylinder	2,3
14	4	LH2 tank lower skirt	2,3,4
15	4	LH2 tank ring	2,3,4
16	8	LH2 tank lower dome	3,4
17	3	Lox tank lower dome	3,4
18	3	Lox tank upper dome	3,4
19	2	Cold Gas Reacting System	3,4
20	8	Engine Thrust Frame 1	3,4
21	1	Engine Thrust Frame 2	3,4
22	1	Engine Thrust Frame 3	3,4
23	3	LH2 feedlines	3,4
24	3	LOX feedlines	3,4
25	4	Helium spheres	3,4,5
26	1	Combustion chamber	3,4,6
27	3	Power unit	3,4
28	1	Turbopump 1	3,4
29	1	Turbopump 2	3,4

Table B.1: List of fragments

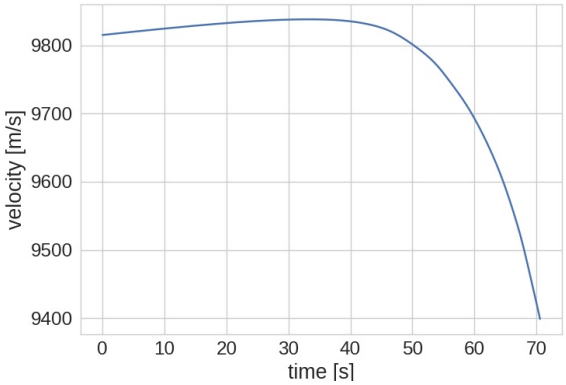


Figure B.3: Time evolution of the Upper Stage velocity before breakup

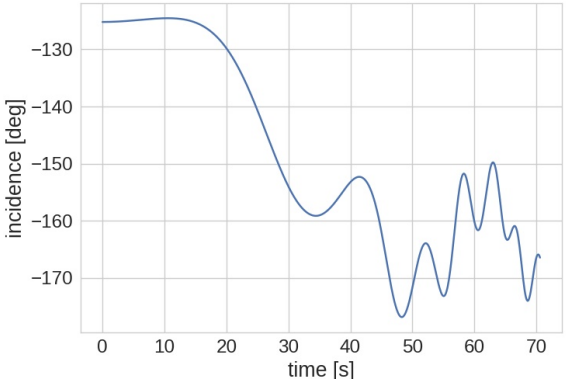


Figure B.4: Time evolution of the Upper Stage incidence before breakup

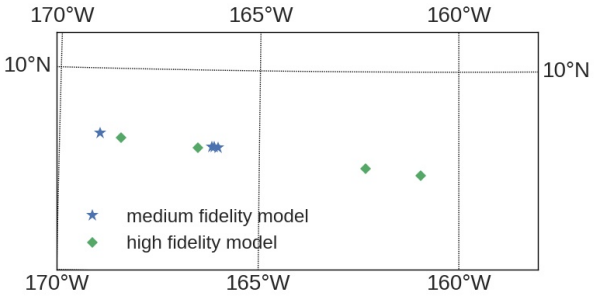


Figure B.5: Impact locations of the medium fidelity model and the high fidelity model.

B.3 Discussion of the two models

The medium fidelity and high fidelity model differ in the fragmentation and the object trajectory before breakup. The high fidelity model uses a 6 DoF trajectory model whereas the medium fidelity model uses a 3 DoF model. Depending on the object, the object attitude motion may

be chaotic especially for objects that are not designed to perform reentry. In these cases, the changing attitude may impact the object aerodynamics and the received heat flux. This in turn affects the trajectory, the thermal response and the breakup.

The thermal response model of the high fidelity prediction is also more refined. First, instead of using correlations from simple shape objects, the convective heat flux is computed for each triangle of the mesh using ARPEGE. Based on those calculations, local thermal responses can be computed for each user-defined zone. Third, within a zone, a 1D thermal response is computed whereas the medium fidelity model makes a uniform temperature assumption.

These improvements directly impact the breakup predictions. Compared to the medium fidelity model where the breakup could only be estimated using a uniform temperature model, the high fidelity model can capture local behaviours depending on the definition of the zones.

Bibliography

[1] H. Klinkrad, *Space debris : models and risk analysis*. Berlin ; New York : Springer ; Chichester, UK : Published in association with Praxis Pub, 2006.

[2] "Esa disco tool." `https://discosweb.esoc.esa.int`. Accessed: 2018-10-30.

[3] E. H. Hirschel, *Basics of Aerothermodynamics*. AIAA Publications, AIAA, Reston, VA, 2005.

[4] J. Anderson Jr, *Hypersonic and High Temperature Gas Dynamics*. AIAA Publications, AIAA, Reston, VA, 2000.

[5] R. N. Cox and L. F. Crabtree, *Elements of Hypersonic Aerodynamics*. Academic Press, 1965.

[6] Y. Prevereaud, *Contribution à la modélisation de la rentrée atmosphérique des débris spatiaux*. PhD thesis, Institut Supérieur de l'Aéronautique et de l'Espace (ISAE), 2014.

[7] A. B. Strong, *Fundamentals of composites manufacturing: materials, methods and applications*. Society of Manufacturing Engineers, 2008.

[8] T. Schleutker, A. Gülhan, T. Lips, E. Kaschnitz, and B. Bonvoisin, "experimental investigations on the demisability of space relevant materials," ESA, 2012.

[9] B. Fritsche, G. Koppenwallner, and T. Lips, "Re-entry analysis for the atv with scarab," tech. rep., HTG Hyperschall-Technologie Gottingen, 2001.

[10] T. Lips, S. Lohle, T. Marynovwsky, D. Rees, H. Stenbeak-Nielsen, M. Beks, and J. Hatton, "Assessment of the ATV-1 Re-entry observation campaign for future re-entry missions," in *Making Safety Matter, Proceedings of the 4th IAASS Conference*, vol. 680 of *ESA Special Publication*, Sept. 2010.

[11] C. Portelli, L. Salotti, L. Anselmo, T. Lips, and A. Tramutola, "Bepposax equatorial uncontrolled re-entry," *Advances in Space Research*, vol. 34, no. 5, pp. 1029 – 1037, 2004. Space Debris.

[12] J. Annaloro, S. Galera, P. Kärräng, G. Prigent, T. Lips, and P. Omaly, "Comparison between two spacecraft-oriented tools: Pampero and scarab," *Journal of Space Safety Engineering*, vol. 4, no. 1, pp. 15 – 21, 2017.

[13] L. Pronzato and W. G. Müller, "Design of computer experiments: space filling and beyond," *Statistics and Computing*, vol. 22, pp. 681–701, May 2012.

[14] "What is space debris, esa faq webpage." `http://www.esa.int/Our_Activities/Operations/Space_Debris/FAQ_Frequently_asked_questions`. Accessed: 2018-10-30.

[15] P. L., "Outer space activities versus outer space," *Journal of Space Law*, vol. 7, no. 2, pp. 115–119, 1979.

[16] "ESA's Annual Space Environment Report," tech. rep., ESA, May 2018.

[17] "Nasa faq webpage." `https://www.nasa.gov/news/debris_faq.html`. Accessed: 2018-11-13.

[18] D. J. Kessler and B. G. Cour-Palais, "Collision frequency of artificial satellites: The creation of a debris belt," *Journal of Geophysical Research: Space Physics*, vol. 83, no. A6, pp. 2637–2646.

[19] H. Klinkrad, B. Fritsche, T. Lips, and G. Koppenwallner, *Re-Entry Prediction and On-Ground Risk Estimation*, pp. 241–288. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006.

[20] N. Dias, "Derivation of the french space operation act requirements in the specifications of the future european launcher ariane 6," *Journal of Space Safety Engineering*, 2018.

[21] "Loi no 2008-518 du 3 juin 2008 relative aux opérations spatiales."

[22] D. J. Kessler, "Collisional cascading: The limits of population growth in low earth orbit," *Advances in Space Research*, vol. 11, pp. 63–66, 1991.

[23] "Satellite Collision Leaves Significant Debris Clouds," *Orbital Debris Quarterly News, NASA Orbital Debris Program Office*, vol. 13, pp. 1–10, 2009.

[24] J. Dolado-Perez, B. Revelin, and R. Di-Costanzo, "Sensitivity analysis of the long-term evolution of the space debris population in leo," *Journal of Space Safety Engineering*, vol. 2, no. 1, pp. 12 – 22, 2015.

[25] J. Dolado-Perez, C. Pardini, and L. Anselmo, "Review of uncertainty sources affecting the long-term predictions of space debris evolutionary models," *Acta Astronautica*, vol. 113, pp. 51 – 65, 2015.

[26] J. Radtke, S. Mueller, V. Schaus, and E. Stoll, "Luca2 - an enhanced long-term utility for collision analysis," in *7th European Conference on Space Debris*, ESA, 2017.

[27] V. Schaus, J. Radtke, E. Stoll, A. Rossi, C. Colombo, S. Tonetti, and I. Holbrough, "Results of reference long-term simulations focussing on passive means to reduce the impact of space debris," in *7th European Conference on Space Debris*, ESA, 2017.

[28] A. Horstmann and E. Stoll, "Investigation of propagation accuracy effects within the modeling of space debris," in *7th European Conference on Space Debris*, ESA, 2017.

[29] J. Radtke and E. Stoll, "Comparing long-term projections of the space debris environment to real world data – looking back to 1990," *Acta Astronautica*, vol. 127, pp. 482 – 490, 2016.

[30] "ESA Space Debris Mitigation Compliance Verification Guidelines," tech. rep., ESA, February 2015.

[31] H. Klinkrad, "Large satellite constellations and related challenges for space debris mitigation," *Journal of Space Safety Engineering*, vol. 4, no. 2, pp. 59 – 60, 2017.

[32] B. B. Virgili, J. Dolado, H. Lewis, J. Radtke, H. Krag, B. Revelin, C. Cazaux, C. Colombo, R. Crowther, and M. Metz, "Risk to space sustainability from large constellations of satellites," *Acta Astronautica*, vol. 126, pp. 154 – 162, 2016. Space Flight Safety.

[33] T. Maury, P. Loubet, J. Ouziel, M. Saint-Amand, L. Dariol, and G. Sonnemann, "Towards the integration of orbital space use in life cycle impact assessment," *Science of The Total Environment*, vol. 595, pp. 642 – 650, 2017.

[34] W. Ailor, W. Hallman, G. Steckel, and M. Weaver, "Analysis of Reentered Debris and Implications for Survivability Modeling," in *4th European Conference on Space Debris* (D. Danesy, ed.), vol. 587 of *ESA Special Publication*, p. 539, Aug. 2005.

[35] B. Fritsche, H. Klinkrad, A. Kashkovsky, and E. Grinberg, "Spacecraft disintegration during uncontrolled atmospheric re-entry," *Acta Astronautica*, vol. 47, no. 2, pp. 513 – 522, 2000. Space an Integral Part of the Information Age.

[36] H. Klinkrad, B. Fritsche, and A. Kashkovsky, "Prediction of Spacecraft Destruction during Uncontrolled Re-Entries," in *Spacecraft Structures, Materials and Mechanical Testing* (C. Stavrinidis, A. Rolfo, and E. Breitbach, eds.), vol. 468 of *ESA Special Publication*, p. 485, 2001.

[37] C. Finzi, C. Bertorello, G. Pinaud, and J.-M. Bouilly, "Simulation of the ariane 5 epc reentry with the fragmentation tool suite," in *7th European Conference on Space Debris*, ESA, 2017.

[38] B. Fritsche, "Uncertainty quantification for re-entry survivability prediction," in *Space Safety is No Accident* (T. Sgobba and I. Rongier, eds.), (Cham), pp. 469–478, Springer International Publishing, 2015.

[39] T. Lips, B. Fritsche, R. Kanzler, T. Schleutker, A. Gülhan, B. Bonvoisin, T. Soares, and G. Sinnema, "About the demisability of propellant tanks during atmospheric re-entry from leo," *Journal of Space Safety Engineering*, vol. 4, no. 2, pp. 99 – 104, 2017.

[40] "Rocket Body Components Survive Reentry," *Orbital Debris Quarterly News, NASA Orbital Debris Program Office*, vol. 2, p. 4, 1997.

[41] "ESA Re-entry Safety Requirements," tech. rep., ESA, December 2017.

[42] "Columbia Crew Survival Investigation Report," tech. rep., NASA/S2008-565, 2008.

[43] A. Soucek, "On-orbit satellite szervicing / close proximity operations legal aspects," in *Cleanspace Industrial Days*, ESA (ESTEC), 2018.

[44] "Arrete du 31 mars 2011 relatif a la reglemnetation technique en application du decret n. 2009-643 du 9 juin 2009 relatif aux autorisations delivrees en application de la loi n 2008-518 du 3 juin 2008 relative aux operations spatiales (nor esrr1103737a)."

[45] P. M. Mehta, M. Kubicek, E. Minisci, and M. Vasile, "Sensitivity analysis and probabilistic re-entry modeling for debris using high dimensional model representation based uncertainty treatment," *Advances in Space Research*, vol. 59, no. 1, pp. 193–211, 2017.

[46] E. M. Alessi, G. Tommei, I. Holbrough, and J. Beck, "Dynamical uncertainty and demisability occurrence for the atmospheric re-entry of soho," *Advances in Space Research*, vol. 62, no. 11, pp. 3033–3047, 2018.

[47] C. Tardioli, M. Kubicek, M. Vasile, E. Minisci, and A. Riccardi, "Comparison of non-intrusive approaches to uncertainty propagation in orbital mechanics," in *AIAA/AAS Astrodynamics Specialist Conference Vail CO*, American Astronautical Society, 2015.

[48] T. Lips, S. Koppenwallner, L. Bianchi, and H. Klingrad, "Risk assessment for destructive reentry," in *5th European Conference on Space Debris*, ESA Special Publication, 2009.

[49] V. Morand, P. Mercier, G. Prigent, E. Bignon, and P. M. Congedo, "CNES activities on polynomial chaos expansion for uncertainty propagation," in *27th AAS/AIAA Space Flight Mechanics Meeting*, (San Antonio, United States), May 2017.

[50] J. Morio and R. Pastel, "Sampling technique for launcher impact safety zone estimation," *Acta Astronautica*, vol. 66, no. 5, pp. 736 – 741, 2010.

[51] J. Morio, R. Pastel, and F. L. Gland, "Missile target accuracy estimation with importance splitting," *Aerospace Science and Technology*, vol. 25, no. 1, pp. 40 – 44, 2013.

[52] J. Morio and M. Balesdent, "Estimation of a launch vehicle stage fallout zone with parametric and non-parametric importance sampling algorithms in presence of uncertain input distributions," *Aerospace Science and Technology*, vol. 52, no. Supplement C, pp. 95 – 101, 2016.

[53] N. Sissenwine, M. Dubin, and H. Wexler, "The us standard atmosphere, 1962," *Journal of Geophysical Research*, vol. 67, no. 9, pp. 3627–3630, 1962.

[54] J. M. Picone, A. E. Hedin, D. P. Drob, and A. C. Aikin, "NRLMSISE-00 empirical model of the atmosphere: Statistical comparisons and scientific issues," *Journal of Geophysical Research (Space Physics)*, vol. 107, p. 1468, Dec. 2002.

[55] T. I. Gombosi, *Gaskinetic theory*. No. 9, Cambridge University Press, 1994.

[56] C. Park, *Nonequilibrium hypersonic aerothermodynamics*. John Wiley and Sons, 1990.

[57] D. C. Wilcox, *Turbulence modelling for CFD*. DCW Industries, La Cañada, 1993.

[58] A. Viladegut, "Enthalpy characterization and assessment of copper catalysis determination in inductively coupled plasma facility," in *45th AIAA Thermophysics Conference*, p. 3107, 2015.

[59] L. Barka, M. Balat-Pichelin, J.-L. Sans, J. Annaloro, and P. Omaly, "Influence of oxidation and emissivity for metallic alloys space debris during their atmospheric entry," in *7th European Conference on Space Debris*, ESA, 2017.

[60] B. Fritsche, "Modelling the Thermal Decomposition of Carbon Fibre Materials During Re-Entry," in *6th European Conference on Space Debris*, vol. 723 of *ESA Special Publication*, p. 60, Aug. 2013.

[61] C. De Persis, *A risk assemssment tool for highly energetic break-up events during the atmospheric re-entry*. PhD thesis, Trinity College Dublin, 2016.

[62] T. Batista Virgili, H. Krag, T. Lips, and E. De Pasquale, "Simulation of the atv re-entry observations," in *Making Safety Matter, Proceedings of the 4th IAASS Conference*, vol. 680 of *ESA Special Publication*, sep 2010.

[63] B. LOS, H. Fraysse, and team Etienne Rapp, "Guide des bonnes pratiques maitrise d'un objet spatial," tech. rep., 2014.

[64] B. Lazare, M. H. Arnal, C. Aussilhou, A. Blazquez, and F. Chemama, "Electra launch and re-entry safety analysis tool," in *Making Safety Matter*, vol. 680 of *ESA Special Publication*, p. 46, Sept. 2010.

[65] E. Doxsey-Whitfield, K. MacManus, S. B. Adamo, L. Pistolesi, J. Squires, O. Borkovska, and S. R. Baptista, "Taking advantage of the improved availability of census data: A first look at the gridded population of the world, version 4," *Papers in Applied Geography*, vol. 1, no. 3, pp. 226–234, 2015.

[66] R. Pastel, *Estimation de probabilités d'évènements rares et de quantiles extrêmes : applications dans le domaine aérospatial*. PhD thesis, 2012. Thèse de doctorat dirigée par Le Gland, François et Morio, Jérôme Mathématiques et applications Rennes 1 2012.

[67] J. N. Garcia, Z. Kutalik, K.-H. Cho, and O. Wolkenhauer, "Level sets and minimum volume sets of probability density functions," *International Journal of Approximate Reasoning*, vol. 34, no. 1, pp. 25 – 47, 2003.

[68] J. Morio, "Non-parametric adaptive importance sampling for the probability estimation of a launcher impact position," *Reliability Engineering and System Safety*, vol. 96, no. 1, pp. 178 – 183, 2011. Special Issue on Safecomp 2008.

[69] C. Parigini, I. Pontijas Fuentes, D. Bonetti, G. Blanco Arnao, and D. Riley, "Debris: an object oriented code for footprint,survivability and risk assessment," in *7th international space debris conference*, ESA, 2018.

[70] C. L. Fèvre, H. Fraysse, V. Morand, A. Lamy, C. Cazaux, P. Mercier, C. Dental, F. Deleflie, and D. Handschuh, "Compliance of disposal orbits with the french space operations act: The good practices and the stela tool," *Acta Astronautica*, vol. 94, no. 1, pp. 234 – 245, 2014.

[71] G. A. Bird, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*. Oxford University Press, 1994.

[72] M. G. Kogan, *Rarefied Gas Dynamics*. Plenum Press, 1969.

[73] G. Koppenwallner, B. Fritsche, T. Lips, and H. Klinkrad, "Scarab -a Multi-Disciplinary Code for Destruction Analysis of Space-Craft during Re-Entry," in *Fifth European Symposium on Aerothermodynamics for Space Vehicles* (D. Danesy, ed.), vol. 563 of *ESA Special Publication*, p. 281, Feb. 2005.

[74] J. Annaloro, P. Omaly, V. Rivola, and M. Spel, "Elaboration of a new spacecraft-oriented tool: Pampero," in *Proceedings of the 8th European Symposium on Aerothermodynamics for Space Vehicles*, 2014.

[75] P. Omaly and M. Spel, "DEBRISK, a Tool for Re-Entry Risk Analysis," in *A Safer Space for Safer World*, vol. 699 of *ESA Special Publication*, p. 70, Jan. 2012.

[76] J. A. Fay, "Theory of stagnation point heat transfer in dissociated air," *Journal of the Aerospace Sciences*, vol. 25, no. 2, pp. 73–85, 1958.

[77] R. D. Klett, "Drag coefficients and heating ratios for right circular cylinders in free-molecular and continuum flow from mach 10 to 30," tech. rep., Sandia Corp., Albuquerque, N. Mex., 1964.

[78] R. Detra, "Generalized heat transfer formulas and graphs for nose cone re-entry into the atmosphere," *ARS Journal*, vol. 31, no. 3, pp. 318–321, 1961.

[79] J. Dobarco-Otero, R. Smith, K. Bledsoe, R. Delaune, W. Rochelle, and N. Johnson, "The object reentry survival analysis tool (orsat)-version 6.0 and its application to spacecraft entry," in *Proceedings of the 56th Congress of the International Astronautical Federation, the International Academy of Astronautics, and International Institute of Space Law, IAC-05-B6*, vol. 3, pp. 17–21, 2005.

[80] J. Annaloro, S. Galera, P. Karrang, Y. Prevereaud, J.-L. Verant, M. Spel, P. Van Hauwaert, and P. Omaly, "Space debris atmospheric entry prediction with spacecraft-oriented tools," in *7th European Conference on Space Debris*, ESA, 2017.

[81] Z. WU, R. HU, X. QU, X. WANG, and Z. WU, "Space debris reentry analysis methods and tools," *Chinese Journal of Aeronautics*, vol. 24, no. 4, pp. 387 – 395, 2011.

[82] J. Annaloro, S. Galera, C. Thiebaut, and P. Omaly, "Latest improvements on the cnes spacecraft-oriented tool: Pampero," in *ESA Industrial Clean Space Days*, ESA, 2018.

[83] M. Frank, M. Weaver, and R. Baker, "A probabilistic paradigm for spacecraft random reentry disassembly," *Reliability Engineering and System Safety*, vol. 90, no. 2, pp. 148 – 161, 2005.

[84] T. Lips, "Re-entry analysis of terrasar-x with scarab," in *54th International Astronautical Congress of the International Astronautical Federation, the International Academy of Astronautics, and the International Institute of Space Law*, pp. IAA–5, 2003.

[85] T. Lips, V. Wartemann, G. Koppenwallner, H. Klinkrad, D. Alwes, J. Dobarco-Otero, R. N. Smith, R. M. Delaune, W. C. Rochelle, and N. L. Johnson, "Comparison of Orsat and Scarab Reentry Survival Results," in *4th European Conference on Space Debris* (D. Danesy, ed.), vol. 587 of *ESA Special Publication*, p. 533, Aug. 2005.

[86] T. Lips, "Equivalent Re-Entry Breakup Altitude and Fragment List," in *6th European Conference on Space Debris*, vol. 723 of *ESA Special Publication*, p. 54, Aug. 2013.

[87] J. Merrifield, J. Beck, G. Markelov, P. Leyland, and R. Molina, "Aerothermal heating methodology in the spacecraft aerothermal model (sam)," in *Space Safety is No Accident* (T. Sgobba and I. Rongier, eds.), (Cham), pp. 463–468, Springer International Publishing, 2015.

[88] J. Beck, J. Merrifield, G. Markelov, I. Holbrough, and R. Molina, "Verification and application of the sam re-entry model," in *Space Safety is No Accident* (T. Sgobba and I. Rongier, eds.), (Cham), pp. 437–443, Springer International Publishing, 2015.

[89] P. T. Summers, Y. Chen, C. M. Rippe, B. Allen, A. P. Mouritz, S. W. Case, and B. Y. Lattimer, "Overview of aluminum alloy mechanical properties during and after fires," *Fire Science Reviews*, vol. 4, p. 3, Apr 2015.

[90] D. Leiser, S. Loehle, F. Zander, R. Choudhury, D. R. Buttsworth, and S. Fasoulas, "Spacecraft material tests under aerothermal and mechanical reentry loads," in *AIAA Scitech 2019 Forum*, AIAA, 2019.

[91] M. Balat-Pichelin, J. Sans, E. Bêche, V. Flaud, and J. Annaloro, "Oxidation and emissivity of inconel 718 alloy as potential space debris during its atmospheric entry," *Materials Characterization*, vol. 127, pp. 379 – 390, 2017.

[92] F. Sanson and C. Frueh, "Quantifying uncertainties in signal position in non-resolved object images: Application to space object observation," *Advances in Space Research*, vol. 63, no. 8, pp. 2436 – 2454, 2019.

[93] F. Sanson and C. Frueh, "Noise quantification in optical observations of resident space objects for probability of detection and likelihood," in *AAS/AIAA Astrodynamic Specialist Conference, Vail, CO*, pp. 15–634, 2015.

[94] F. Sanson, N. Villedieu, F. Panerai, O. Chazot, P. M. Congedo, and T. E. Magin, "Quantification of uncertainty on the catalytic property of reusable thermal protection materials from high enthalpy experiments," *Experimental Thermal and Fluid Science*, vol. 82, pp. 414 – 423, 2017.

[95] F. Sanson, F. Panerai, T. E. Magin, and P. M. Congedo, "Robust reconstruction of the catalytic properties of thermal protection materials from sparse high-enthalpy facility experimental data," *Experimental Thermal and Fluid Science*, vol. 96, pp. 482 – 492, 2018.

[96] R. C. Smith, *Uncertainty quantification: theory, implementation, and applications*, vol. 12. Siam, 2013.

[97] G. Iaccarino, "Uncertainty quantification in computational science," 2014.

[98] G. Shafer, *A mathematical theory of evidence*, vol. 42. Princeton university press, 1976.

[99] F. P. A. Coolen, "On the use of imprecise probabilities in reliability," *Quality and Reliability Engineering International*, vol. 20, no. 3, pp. 193–202, 2004.

[100] R. Schöbi and B. Sudret, "Uncertainty propagation of p-boxes using sparse polynomial chaos expansions," *Journal of Computational Physics*, vol. 339, pp. 307–327, June 2017.

[101] B. De Finetti, *"Theory of Probability: A critical introductory treatment"*. Wiley, 1990.

[102] O. P. Le Maître and O. M. Knio, *Spectral Methods for Uncertainty Quantification*. Scientific Computation, Dordrecht: Springer Netherlands, 2010.

[103] R. E. Bellman and S. E. Dreyfus, *Applied dynamic programming*. Princeton university press, 2015.

[104] R. E. Caflisch, "Monte carlo and quasi-monte carlo methods," *Acta Numerica*, vol. 7, p. 1–49, 1998.

[105] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.

[106] M. Stein, "Large sample properties of simulations using latin hypercube sampling," *Technometrics*, vol. 29, no. 2, pp. 143–151, 1987.

[107] D. Xiu and G. E. Karniadakis, "The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations," *SIAM Journal on Scientific Computing*, vol. 24, pp. 619–644, Jan. 2002.

[108] R. G. Ghanem and P. D. Spanos, *Stochastic Finite Elements: A Spectral Approach*. Berlin, Heidelberg: Springer-Verlag, 1991.

[109] W. Gautschi, *Orthogonal Polynomials: Quadrature , and Approximation*. Clarendon Press, Oxford, 2004.

[110] M. T. Reagan, H. N. Najm, R. G. Ghanem, and O. M. Knio, "Uncertainty quantification in reacting-flow simulations through non-intrusive spectral projection," *Combustion and Flame*, vol. 132, no. 3, pp. 545 – 555, 2003.

[111] P. Congedo, C. Corre, and J.-M. Martinez, "Shape optimization of an airfoil in a bzt flow with multiple-source uncertainties," *Computer Methods in Applied Mechanics and Engineering*, vol. 200, no. 1, pp. 216 – 232, 2011.

[112] G. Blatman and B. Sudret, "Adaptive sparse polynomial chaos expansion based on least angle regression," *Journal of Computational Physics*, vol. 230, no. 6, pp. 2345 – 2367, 2011.

[113] D. Xiu, *Numerical methods for Stochastic Computations: a spectral method approach*. Princeton University Press, 2010.

[114] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*, vol. 1. Springer series in statistics New York, 2001.

[115] K. Tang, P. M. Congedo, and R. Abgrall, "Adaptive surrogate modeling by anova and sparse polynomial dimensional decomposition for global sensitivity analysis in fluid simulation," *Journal of Computational Physics*, vol. 314, pp. 557 – 589, 2016.

[116] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[117] R. K. Tripathy and I. Bilionis, "Deep uq: Learning deep neural network surrogate models for high dimensional uncertainty quantification," *Journal of Computational Physics*, vol. 375, pp. 565 – 588, 2018.

[118] S. Karumuri, R. Tripathy, I. Bilionis, and J. Panchal, "Simulator-free Solution of High-Dimensional Stochastic Elliptic Partial Differential Equations using Deep Neural Networks," *arXiv e-prints*, p. arXiv:1902.05200, Feb 2019.

[119] N. Cressie, *Statistics for spatial data*. John Wiley and Sons, 2015.

[120] C. E. Rasmussen, "Gaussian processes for machine learning," MIT Press, 2006.

[121] I. Bilionis and N. Zabaras, "Multi-output local gaussian process regression: Applications to uncertainty quantification," *Journal of Computational Physics*, vol. 231, no. 17, pp. 5718–5746, 2012.

[122] J. E. Oakley and A. O'Hagan, "Probabilistic sensitivity analysis of complex models: a bayesian approach," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 66, no. 3, pp. 751–769, 2004.

[123] A. F. Cortesi, G. Jannoun, and P. M. Congedo, "Kriging-sparse polynomial dimensional decomposition surrogate model with adaptive refinement," *Journal of Computational Physics*, vol. 380, pp. 212 – 242, 2019.

[124] R. Tripathy, I. Bilionis, and M. Gonzalez, "Gaussian processes with built-in dimensionality reduction: Applications to high-dimensional uncertainty propagation," *Journal of Computational Physics*, vol. 321, pp. 191 – 223, 2016.

[125] P. G. Constantine, E. Dow, and Q. Wang, "Active subspace methods in theory and practice: applications to kriging surfaces," *SIAM Journal on Scientific Computing*, vol. 36, no. 4, pp. A1500–A1524, 2014.

[126] K. Konakli and B. Sudret, "Polynomial meta-models with canonical low-rank approximations: Numerical insights and comparison to sparse polynomial chaos expansions," *Journal of Computational Physics*, vol. 321, pp. 1144 – 1169, 2016.

[127] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola, *Global sensitivity analysis: the primer*. John Wiley and Sons, 2008.

[128] B. Iooss and P. Lemaître, *A Review on Global Sensitivity Analysis Methods*, pp. 101–122. Boston, MA: Springer US, 2015.

[129] I. Sobol, "Theorems and examples on high dimensional model representation," *Reliability Engineering and System Safety*, vol. 79, no. 2, pp. 187 – 193, 2003. SAMO 2001: Methodological advances and innovative applications of sensitivity analysis.

[130] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola, "Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index," *Computer Physics Communications*, vol. 181, no. 2, pp. 259 – 270, 2010.

[131] A. Saltelli, "Making best use of model evaluations to compute sensitivity indices," *Computer Physics Communications*, vol. 145, no. 2, pp. 280 – 297, 2002.

[132] F. Gamboa, A. Janon, T. Klein, A. Lagnoux, and C. Prieur, "Statistical inference for sobol pick-freeze monte carlo method," *Statistics*, vol. 50, no. 4, pp. 881–902, 2016.

[133] A. Owen and C. Prieur, "On shapley value for measuring importance of dependent inputs," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 5, no. 1, pp. 986–1002, 2017.

[134] G. Chastaing, F. Gamboa, and C. Prieur, "Generalized sobol sensitivity indices for dependent variables: numerical methods," *Journal of Statistical Computation and Simulation*, vol. 85, no. 7, pp. 1306–1333, 2015.

[135] F. Gamboa, A. Janon, T. Klein, and A. Lagnoux, "Sensitivity indices for multivariate outputs," *Comptes Rendus Mathematique*, vol. 351, no. 7, pp. 307 – 310, 2013.

[136] W. A. Huber, ""why risk is not variance: An expository note"," *Risk Analysis*, vol. 30, no. 3, pp. 327–328, 2010.

[137] L. A. T. Cox, Jr, "Why risk is not variance: An expository note," *Risk Analysis*, vol. 28, no. 4, pp. 925–928, 2008.

[138] E. Borgonovo, W. Castaings, and S. Tarantola, "Moment independent importance measures: New results and analytical test cases," *Risk Analysis*, vol. 31, no. 3, pp. 404–428, 2011.

[139] E. Borgonovo, W. Castaings, and S. Tarantola, "Model emulation and moment-independent sensitivity analysis: An application to environmental modelling," *Environmental Modelling and Software*, vol. 34, pp. 105 – 115, 2012. Emulation techniques for the reduction and sensitivity analysis of complex environmental models.

[140] S. D. Veiga, "Global sensitivity analysis with dependence measures," *Journal of Statistical Computation and Simulation*, vol. 85, no. 7, pp. 1283–1305, 2015.

[141] F. Gamboa, T. Klein, and A. Lagnoux, "Sensitivity analysis based on cramér–von mises distance," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 6, no. 2, pp. 522–548, 2018.

[142] J.-C. Fort, T. Klein, and N. Rachdi, "New sensitivity analysis subordinated to a contrast," *Communications in Statistics - Theory and Methods*, vol. 45, no. 15, pp. 4349–4364, 2016.

[143] R. Schobi, B. Sudret, and J. Wiart, "Polynomial-chaos-based kriging," *International Journal for Uncertainty Quantification*, vol. 5, no. 2, pp. 171–193, 2015.

[144] C. C. Aggarwal, *Data classification: algorithms and applications*. CRC press, 2014.

[145] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in large margin classifiers*, pp. 61–74, MIT Press, 1999.

[146] C. K. I. Williams and D. Barber, "Bayesian classification with gaussian processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 1342–1351, Dec 1998.

[147] T. P. Minka, *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology, 2001.

[148] GPy, "GPy: A gaussian process framework in python." `http://github.com/SheffieldML/GPy`, since 2012.

[149] N. Cressie, "The origins of kriging," *Mathematical Geology*, vol. 22, no. 3, pp. 239–252, 1990.

[150] M. Stein, *Interpolation of spatial data: some theory for kriging*. Springer Science and Business Media, 2012.

[151] T. A. Plate, "Accuracy versus interpretability in flexible modeling: Implementing a tradeoff using gaussian process models," *Behaviormetrika*, vol. 26, no. 1, pp. 29–50, 1999.

[152] W. Betz, I. Papaioannou, and D. Straub, "Numerical methods for the discretization of random fields by means of the Karhunen-Loève expansion," *Computer Methods in Applied Mechanics and Engineering*, vol. 271, pp. 109–129, Apr. 2014.

[153] G. Matheron, "The intrinsic random functions and their applications," *Advances in Applied Probability*, vol. 5, no. 3, pp. 439–468, 1973.

[154] D. Krige, "A statistical approach to some basic mine valuation problems on the witwatersrand," *J. of the Chem., Metal. and Mining Soc. of South Africa*, vol. 52, no. 6, p. 119–139, 1951.

[155] G. Perrin, C. Soize, S. Marque-Pucheu, and J. Garnier, "Nested polynomial trends for the improvement of gaussian process-based predictors," *Journal of Computational Physics*, vol. 346, pp. 389 – 402, 2017.

[156] L. Pronzato, "Minimax and maximin space-filling designs : some properties and methods for construction," in *Societe Francaise de Statistique*, 2017.

[157] M. D. McKay, R. J. Beckman, and W. J. Conover, "Comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.

[158] B. Iooss, L. Boussouf, V. Feuillard, and A. Marel, "Numerical studies of the metamodel fitting and validation processes," *International Journal on Advances in Systems and Measurements*, vol. 3, pp. 11–21, 2010.

[159] E. R. Van Dam, B. Husslage, D. Den Hertog, and H. Melissen, "Maximin latin hypercube designs in two dimensions," *Operations Research*, vol. 55, no. 1, pp. 158–169, 2007.

[160] B. Tang, "Orthogonal array-based latin hypercubes," *Journal of the American Statistical Association*, vol. 88, no. 424, pp. 1392–1397, 1993.

[161] P. J. Zemroch, "Cluster analysis as an experimental design generator, with application to gasoline blending experiments," *Technometrics*, vol. 28, no. 1, pp. 39–49, 1986.

[162] B. A. Flury, "Principal points," *Biometrika*, vol. 77, pp. 33–41, 03 1990.

[163] R. Lekivetz and B. Jones, "Fast flexible space-filling designs for nonrectangular regions," *Quality and Reliability Engineering International*, vol. 31, no. 5, pp. 829–837, 2015.

[164] P. Gilles, "Optimal vector quantization: from signal processing to clustering and numerical probability," CEMRACS, 2017.

[165] S. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, pp. 129–137, March 1982.

[166] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological Cybernetics*, vol. 43, pp. 59–69, Jan 1982.

[167] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global optimization*, vol. 13, no. 4, pp. 455–492, 1998.

[168] B. Echard, N. Gayton, M. Lemaire, and N. Relun, "A combined importance sampling and kriging reliability method for small failure probabilities with time-demanding numerical models," *Reliability Engineering and System Safety*, vol. 111, no. Supplement C, pp. 232 – 240, 2013.

[169] D. Bursztyn and D. M. Steinberg, "Comparison of designs for computer experiments," *Journal of Statistical Planning and Inference*, vol. 136, no. 3, pp. 1103 – 1119, 2006.

[170] K. Ritter, *Average-case analysis of numerical problems*. Springer, 2007.

[171] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," *Statistical science*, pp. 409–423, 1989.

[172] C. Currin, T. Mitchell, M. Morris, and D. Ylvisaker, "Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments," *Journal of the American Statistical Association*, vol. 86, no. 416, pp. 953–963, 1991.

[173] T. J. Mitchell, "An algorithm for the construction of "d-optimal" experimental designs," *Technometrics*, vol. 16, no. 2, pp. 203–210, 1974.

[174] M. C. Shewry and H. P. Wynn, "Maximum entropy sampling," *Journal of Applied Statistics*, vol. 14, no. 2, pp. 165–170, 1987.

[175] J. Sacks, S. B. Schiller, and W. J. Welch, "Designs for computer experiments," *Technometrics*, vol. 31, no. 1, pp. 41–47, 1989.

[176] D. J. C. MacKay, "Information-based objective functions for active data selection," *Neural Computation*, vol. 4, no. 4, pp. 590–604, 1992.

[177] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *Journal of Machine Learning Research (JMLR)*, vol. 9, pp. 235–284, February 2008.

[178] A. Gorodetsky and Y. Marzouk, "Mercer kernels and integrated variance experimental design: Connections between gaussian process regression and polynomial approximation," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 4, no. 1, pp. 796–828, 2016.

[179] N. Ramakrishnan, C. Bailey-Kellogg, S. Tadepalli, and V. N. Pandey, "Gaussian processes for active data mining of spatial aggregates," in *Proceedings of the 2005 SIAM International Conference on Data Mining*, pp. 427–438, SIAM, 2005.

[180] J. Beck and S. Guillas, "Sequential design with mutual information for computer experiments (mice): emulation of a tsunami model," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 4, no. 1, pp. 739–766, 2016.

[181] B. Gauthier and L. Pronzato, "Approximation of imse-optimal designs via quadrature rules and spectral decomposition," *Communications in Statistics - Simulation and Computation*, vol. 45, no. 5, pp. 1600–1612, 2016.

[182] D. A. Cohn, "Neural network exploration using optimal experiment design," in *Advances in Neural Information Processing Systems 6* (J. D. Cowan, G. Tesauro, and J. Alspector, eds.), pp. 679–686, Morgan-Kaufmann, 1994.

[183] R. B. Gramacy and H. K. H. Lee, "Adaptive design and analysis of supercomputer experiments," *Technometrics*, vol. 51, no. 2, pp. 130–145, 2009.

[184] B. Settles, "Active learning literature survey," tech. rep., 2010.

[185] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *SIGIR'94*, pp. 3–12, Springer, 1994.

[186] N. Roy and A. McCallum, "Toward optimal active learning through sampling estimation of error reduction," in *ICML*, 2001.

[187] T. M. Hospedales, S. Gong, and T. Xiang, "Finding rare classes: Active learning with generative and discriminative models," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, pp. 374–386, Feb 2013.

[188] D. Pelleg and A. W. Moore, "Active learning for anomaly and rare-category detection," in *Advances in neural information processing systems*, pp. 1073–1080, 2005.

[189] S. Amaral, D. Allaire, and K. Willcox, "A decomposition-based approach to uncertainty analysis of feed-forward multicomponent systems," *International Journal for Numerical Methods in Engineering*, vol. 100, no. 13, pp. 982–1005, 2014.

[190] P. G. Constantine and E. T. Phipps, "A lanczos method for approximating composite functions," *Applied Mathematics and Computation*, vol. 218, no. 24, pp. 11751–11762, 2012.

[191] P. G. Constantine, E. T. Phipps, and T. M. Wildey, "Efficient uncertainty propagation for network multiphysics systems," *International Journal for Numerical Methods in Engineering*, vol. 99, no. 3, pp. 183–202, 2014.

[192] M. Arnst, R. Ghanem, E. Phipps, and J. Red-Horse, "Dimension reduction in stochastic modeling of coupled problems," *International Journal for Numerical Methods in Engineering*, vol. 92, no. 11, pp. 940–968, 2012.

[193] S. Dubreuil, N. Bartoli, C. Gogu, T. Lefebvre, and J. M. Colomer, "Extreme value oriented random field discretization based on an hybrid polynomial chaos expansion — kriging approach," *Computer Methods in Applied Mechanics and Engineering*, vol. 332, pp. 540 – 571, 2018.

[194] S. Dubreuil, N. Bartoli, T. Lefebvre, and C. Gogu, "Efficient global multidisciplinary optimization based on surrogate models," in *Multidisciplinary Analysis and Optimization Conference*, pp. 367–377, AIAA AVIATION Forum, (AIAA 2018-3745), 2018.

[195] X. Chen, B. Ng, Y. Sun, and C. Tong, "A flexible uncertainty quantification method for linearly coupled multi-physics systems," *Journal of Computational Physics*, vol. 248, pp. 383–401, 2013.

[196] A. Mittal, X. Chen, C. Tong, and G. Iaccarino, "A flexible uncertainty propagation framework for general multiphysics systems," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 4, no. 1, pp. 218–243, 2016.

[197] S. Sankararaman, *Uncertainty quantification and integration in engineering systems*. PhD thesis, Vanderbilt University, 2012.

[198] A. C. Damianou and N. D. Lawrence, "Deep gaussian processes.," in *AISTATS*, pp. 207–215, 2013.

[199] A. Girard, C. E. Rasmussen, J. Quinonero-Candela, R. Murray-Smith, O. Winther, and J. Larsen, "Multiple-step ahead prediction for non linear dynamic systems—a gaussian process treatment with propagation of the uncertainty," *Advances in neural information processing systems*, vol. 15, pp. 529–536, 2003.

[200] S. Marque-Pucheu, G. Perrin, and J. Garnier, "Efficient sequential experimental design for surrogate modeling of nested codes," *ArXiv e-prints*, Dec. 2017.

[201] I. Sobol', "On the distribution of points in a cube and the approximate evaluation of integrals," *USSR Computational Mathematics and Mathematical Physics*, vol. 7, no. 4, pp. 86 – 112, 1967.

[202] T. J. Santner, B. J. Williams, and W. I. Notz, *The design and analysis of computer experiments*. Springer Science and Business Media, 2013.

[203] C.-W. Ko, J. Lee, and M. Queyranne, "An exact algorithm for maximum entropy sampling," *Operations Research*, vol. 43, no. 4, pp. 684–691, 1995.

[204] R. Storn and K. V. Price, "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optimization*, vol. 11, pp. 341–359, 1997.

[205] T. Ishigami and T. Homma, "An importance quantification technique in uncertainty analysis for computer models," in *[1990] Proceedings. First International Symposium on Uncertainty Modeling and Analysis*, pp. 398–403, Dec 1990.

[206] C. Bertorello, C. Finzi, G. Pinaud, L. Rhidane, and J.-M. Bouilly, "Adryans® v5.0 – a 1-dimension object oriented survivability tool," in *7th European Conference on Space Debris*, ESA, 2017.

[207] D. Delorme and S. Lirola, "Caracteristation de l' environement naturel pour un lanceur spatial decollant du centre spatial guyanais (csg)," tech. rep., CNES, 2013.

[208] T. E. Oliphant, "A bayesian perspective on estimating mean, variance, and standard-deviation from data," 2006.

[209] P. Brémaud, *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*, vol. 31. Springer Science & Business Media, 2013.

[210] C. F. F. Karney, "Algorithms for geodesics," *Journal of Geodesy*, vol. 87, pp. 43–55, Jan 2013.

[211] F. Cadini, F. Santos, and E. Zio, "An improved adaptive kriging-based importance technique for sampling multiple failure regions of low probability," *Reliability Engineering & System Safety*, vol. 131, pp. 109–117, 2014.

[212] B. Echard, N. Gayton, and M. Lemaire, "Ak-mcs: An active learning reliability method combining kriging and monte carlo simulation," *Structural Safety*, vol. 33, no. 2, pp. 145 – 154, 2011.

[213] V. Dubourg, B. Sudret, and F. Deheeger, "Metamodel-based importance sampling for structural reliability analysis," *Probabilistic Engineering Mechanics*, vol. 33, pp. 47–57, 2013.

[214] N. Razaaly and P. M. Congedo, "Novel algorithm using active metamodel learning and importance sampling: Application to multiple failure regions of low probability," *Journal of Computational Physics*, vol. 368, pp. 92 – 114, 2018.

[215] B. Fritsche and S. Stemmens, "Case studies for uncertainty quantification of a high-fidelityspacecraft oriented break-up tool," in *8th Europeans Symposium on Aerothermodynamics for Space Vehicles*, ESA, 2015.

[216] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, pp. 148–175, Jan 2016.

[217] P. Hennig and C. J. Schuler, "Entropy search for information-efficient global optimization," *Journal of Machine Learning Research*, vol. 13, no. Jun, pp. 1809–1837, 2012.

[218] W. Vesely, F. Goldberg, N. Roberts, and D. Haasl, "Fault tree handbook," 1981.

[219] C. M. Bishop *et al.*, *Neural networks for pattern recognition*. Oxford university press, 1995.

[220] M. Raissi, "Deep hidden physics models: Deep learning of nonlinear partial differential equations," *Journal of Machine Learning Research*, vol. 19, no. 25, pp. 1–24, 2018.

[221] M. Raissi and G. E. Karniadakis, "Hidden physics models: Machine learning of nonlinear partial differential equations," *Journal of Computational Physics*, vol. 357, pp. 125 – 141, 2018.

[222] C. Park, J. Z. Huang, and Y. Ding, "A computable plug-in estimator of minimum volume sets for novelty detection," *Operations Research*, vol. 58, no. 5, pp. 1469–1480, 2010.

[223] G. Allaire, *Shape optimization by the homogenization method*, vol. 146. Springer Science & Business Media, 2012.

[224] F. W. Matting, "Approximate bridging relations in the transitional regime between continuum and free-molecule flows," *Journal of Spacecraft and Rockets*, vol. 8, no. 1, pp. 35–40, 1971.

Titre : Estimation du risque humain lié à la retombée d'objets spatiaux sur Terre

Mots Clefs : Rentrée atmosphérique ; Quantification des Incertitudes ; Débris Spatiaux

Résumé : Les réglementations récentes imposent la rentrée dans l'atmosphère d'objects spatiaux artificiels en fin de vie et la quantification du risque humain associé. Le calcul du risque repose sur des simulations numériques de la rentrée de l'objet. Ces simulations multi-physiques utilisent des modèles mathématiques simplistes et se basent sur une connaissance imparfaite des conditions de rentrée. Dans ce travail, nous proposons de nouvelles techniques de quantification d'incertitude pour améliorer la robustesse des prédictions de rentrée. Dans un premier temps, un simulateur de rentrée est assemblé. La prédiction de la rentrée d'un objet étant un problème multi-physique complexe, notre simulateur utilise un système de solveurs chaînés dans lequel chaque solveur simule une phase de la rentrée avec sa physique particulière. Nous développons ensuite deux stratégies pour réaliser la propagation d'incertitude à faible cout dans notre simulateur. Ces deux stratégies s'appuient sur des modèles de substitutions.

La première stratégie de substitution de modèle est une approche générique permettant d'approcher un système de solveurs chaînés par un système de processus gaussiens (System of Gaussian Processes, SoGP). Cette approximation probabiliste est construite en représentant chaque solveur (ou groupe de solveurs) par un processus gaussien (GP). Nous montrons que la variance prédictive du SoGP est décomposable en contributions associées à chaque GP. Cette décomposition de la variance est ensuite exploitée pour concevoir des stratégies d'apprentissage actif et générer des ensembles d'entrainement parcimonieux en renforçant l'apprentissage du GP le moins fiable. La performance du SoGP est étudiée sur plusieurs cas analytiques et industriels. Dans toutes les situations considérées, le SoGP surpasse les approches plus directes.

La seconde contribution de la thèse porte sur la construction de modèle de substitution pour la prédiction de la survie d'objets spatiaux. Lors de la rentrée, un objet spatial se fragmente et génère des débris. Certains débris brulent dans l'atmosphère tandis que d'autres atteignent le sol. Pour évaluer la survie d'un fragment, il faut déterminer s'il atteint le sol et, si c'est le cas, évaluer son point de chute et le risque associé. Dans ce travail, on propose une formulation originale du problème de la survie pour calculer efficacement le risque avec un modèle de substitution. Ce modèle de substitution repose sur la composition d'un classificateur et d'un processus gaussien. Le modèle est entraîné à l'aide d'une stratégie d'apprentissage actif dédiée qui équilibre les contributions du classificateur et du GP à l'erreur de prédiction de la survie, en proposant des plans d'entrainement adaptés.

Pour finir, les méthodes proposées dans la thèse sont appliquées à la simulation de la rentrée contrôlée d'un étage supérieur de fusée et la survie des fragments qui en résulte. Un grand nombre d'incertitudes (38) sont prises en compte et propagées, comprenant les caractéristiques de l'orbite initiale, les conditions de désorbitation, les propriétés du matériau de l'étage supérieur, les entrées du modèle d'atmosphère et les propriétés des matériaux des fragments. De plus, un modèle probabiliste de fragmentation est utilisé pour prédire de manière robuste la rupture de l'objet en tenant compte des incertitudes de modélisation. Les méthodes développées dans la thèse permettent d'estimer à un coût de calcul raisonnable les statistiques des conditions de vol au moment de la fragmentation, la probabilité de survie de chaque fragment et le risque humain associé. Une analyse de sensibilité globale montre que les incertitudes les plus influentes sont liées au modèle de fragmentation et aux conditions de désorbitation. Cette étude démontre ainsi la capacité de notre simulateur à produire une mesure robuste du risque au sol, sur un scénario de rentrée réaliste, et à un coût numérique acceptable.

Title : Reentry Prediction of Man-made Space Object under Uncertainties and on-ground Risk Assessment

Keys words : Atmosphere reentry ; Uncertainty Quantification ; Space Debris

Abstract : Recent regulations impose the re-entry of human-made end-of-life space object with a rigorous assessment of the risk for human assets. The risk evaluation requires sequences of complex numerical simulations accounting for the multi-physics phenomena occurring during the reentry of a space object, e.g., fluid-structure interactions and heat transfer. Further, these simulations are inaccurate because they rely on overly simplified models and partial knowledge of the reentry conditions.

In this thesis, we propose novel uncertainty quantification techniques to deal with some of the uncertainties characterizing the problem and apply them to predict the risk for human assets due to the reentry of a space object.

First, we construct a system of solvers to predict both the controlled or uncontrolled reentry of space objects. Compared to the existing reentry software, our system naturally accommodates the uncertainty in the object breakup predictions. Moreover, the constitutive solvers are interfaced and coupled within a framework that allows a single user to perform parallel runs of the full system.

Second, we present two original methods to propagate the uncertainties in reentry predictions using the system of solvers. First, we construct a surrogate model approximating the directed systems of solvers, using a system of Gaussian Processes (SoGP). We build this probabilistic surrogate by approximating each solver (or a group of solvers) of the directed system by a Gaussian Process (GP). We show that the predictive variance of the SoGP is composed of individual contributions from each GP. We use this decomposition of the variance decomposition to develop active learning strategies based on training datasets wich are enriched parsimoniously to improve the prediction of the least reliable GP only. We assessed the performance of the SoGP on several analytical and industrial cases. The SoGP coupled with active learning strategies yielded systematically significant improvements.

The second method aims at predicting the survivability of space objects. During a space reentry event, the object can break up and generate fragments. Some fragments desintegrate in the atmosphere while others survive to the ground. Assessing the survivability of a fragment implies determining whether it reaches the ground or not and if it does, the impact location and the risk associated. We propose an original formulation of the survivability assessment problem to efficiently estimate the risk. The proposed method involves the composition of a classifier (demise prediction) with a Gaussian Process (impact location prediction). Dedicated active learning strategies are designed to balance the prediction errors of the classifier and GP and allocate training samples adequately.

Finally, we apply the methods developed in the thesis to the prediction of the controlled reentry of a rocket upper-stage. The problem involves a large number of uncertainties (38), including the initial orbit properties, the deorbiting conditions, the upper stage material characteristics, the atmosphere model parameters, and the fragment material uncertainties. Moreover, we use a probabilistic breakup model for the object breakup to account for the model uncertainties. With our methods, we estimate at a reasonable computational cost the statistics of the conditions at breakup, the survival probability of the fragments, the casualty area, and the human risk. Global sensitivity analyses of the breakup conditions and casualty area provide a ranking of the most critical uncertainties. This study demonstrates the capability of our surrogate simulator to produce a robust measure of on-ground risk for a realistic reentry scenario.

Université Paris-Saclay Espace Technologique Bâtiment Discovery
Route de l'Orme aux Merisiers - RD 128
91190 Saint-Aubin