Contribution à la conception d'accélérateurs matériels pour systèmes autonomes intelligents

Catherine Dezan

HABILITATION À DIRIGER DES RECHERCHES

L'UNIVERSITE
DE BRETAGNE OCCIDENTALE

ECOLE DOCTORALE N° 601
MAthématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : Informatique

Par
Catherine DEZAN

« Contribution à la conception d'accélérateurs matériels pour systèmes autonomes intelligents »

Présentée et soutenue à l'UBO, Brest, le 8 septembre 2022
Unité de recherche : Lab-STICC, UMR 6285

Rapporteurs avant soutenance :

Pierre Boulet	Professeur des Universités, Université de Lille
Emmanuel Casseau	Professeur des Universités, Université de Rennes
Olivier Romain	Professeur des Universités, Cergy Paris Université

Composition du Jury :

Rapporteurs :	Pierre Boulet	Professeur des Universités, Université de Lille
	Emmanuel Casseau	Professeur des Universités, Université de Rennes
	Olivier Romain	Professeur des Universités, Cergy Paris Université
Examinateurs :	Jean-Philippe Babau	Professeur des Universités, Université de Brest
	Jean-Philippe Diguet	Directeur de recherche CNRS, IRL CROSSING, Adélaide, Australie
	Frank Singhoff	Professeur des Universités, Université de Brest

Remerciements

Je tiens d'abord à remercier les membres du jury d'avoir accepté d'évaluer ce travail, et de me donner l'opportunité de le défendre. Je transmets mes sincères remerciements à Pierre Boulet, Emmanuel Casseau, Olivier Romain d'avoir accepté d'être rapporteurs ainsi qu'à Jean-Philippe Babau, Frank Singhoff et jean-Philippe Diguet pour leur présence dans ce jury.

Je tiens aussi à remercier les personnes qui ont participé aux travaux de recherches abordés dans ce document.
Tout d'abord les étudiant(e)s que j'ai pu suivre lors de leur doctorat et qui m'ont fortement inspiré. Je leur souhaite une bonne continuation dans la voie des sciences, en espérant qu'ils(elles) trouvent à leur tour aussi leurs voies et leurs inspirations.

J'adresse aussi mes remerciements à mes collègues, et collaborateurs avec qui j'ai eu plaisir d'innover, d'explorer afin de satisfaire notre curiosité et soif de connaissance. Merci à David, Laurent, Stéphane, Erwan, Jalil, Frank, Jean-Philippe, Alain, Patrick, Dominique, Thierry, Philippe, Emmanuel, Loic, Bernard... et aussi à Luis, Duncan, Kalinka qui m'ont accompagné dans les projets de recherches outre-mer en y apportant beaucoup de chaleur humaine.
Spécial merci à Jean-Philippe Diguet et à Gilles Coppin pour le chemin parcouru ensemble.

"Ne crains pas d'avancer lentement, crains seulement de t'arrêter".
Proverbe chinois

sans oublier avec un sourire que

"Un chemin qui descend est un chemin qui monte
en sens inverse et réciproquement".
Pierre Dac

Résumé

L'intelligence artificielle (IA) est entrée dans notre quotidien. L'IA est omniprésente via les systèmes de recommandations de notre navigateur web, via nos objets connectés, notre maison connectée, via nos téléphones portables...
Dans ce document, on s'intéresse à l'IA embarquée et notamment aux accélérateurs matériels qui peuvent être mis en place dans le cadre des véhicules autonomes. Trois aspects y sont plus particulièrement traités : 1) le cadre méthodologique pour la conception des accélérateurs matériels, 2) la sûreté de fonctionnement des drones autonomes et 3) la mise en place de la décision pour la planification de mission. Pour la conception des accélérateurs matériels, de nouveaux flots de conception sont proposés afin faciliter l'intégration de nouvelles spécifications ainsi que leur compilation vers des supports matériels de type FPGA. Les études menées dans le cadre de la sûreté de fonctionnement ont permis de montrer comment le modèle des réseaux Bayésiens pouvait contribuer au diagnostic et à la gestion des défaillances du système, et comment il pouvait être aussi embarquable sur le drone. Enfin on s'intéresse au mécanisme de décision d'une planification de mission en y intégrant ces éléments de diagnostic pour assurer une autonomie complète du drone. Ces travaux ont permis de dresser un bilan et quelques perspectives sur les méthodes et méthodologies de conception d'accélérateurs matériels envisagées pour la gestion de missions de drones autonomes.

Mots-clés— Accélérateurs matériels, IA embarquée, Diagnostic et Décision embarquée, Véhicules autonomes

Table des matières

Chapitre 1

Introduction

1.1 Contexte scientifique

Dans ce manuscrit, nous nous intéressons au domaine de l'Intelligence Artificielle Embarquée (IAE) qui est un domaine en pleine émergence depuis quelques années. Ce domaine a pris son essor grâce à deux facteurs essentiels :

— l'avancée actuelle des techniques d'apprentissage autour des réseaux de neurones avec le Deep Learning

— l'avancée technologique des plateformes supportant le calcul intensif pouvant être actuellement embarquées

Dans ce chapitre, nous nous proposons d'aborder brièvement ces deux aspects en analysant l'état actuel de cette tendance et les verrous restants, en ciblant plus particulièrement le domaine des véhicules autonomes. Les contributions scientifiques et motivations de ces travaux sont ensuite posées et développées plus en détail dans les différents chapitres du manuscrit. Le chapitre 2 est dédié aux méthodes et outils nécessaires à la conception d'architectures embarquées ciblant plus particulièrement les supports reconfigurables. Le chapitre 3 aborde le sujet de l'intégration de modèles probabilistes pour assurer le diagnostic de défaillance dans le cadre de véhicules autonomes. Le chapitre 4 traite de la problématique de la décision et de l'apprentissage embarqué dans le cadre de planification de mission. En conclusion, une perspective de ces travaux est proposée à court et moyen termes dans le chapitre 5.

1.1.1 IA embarquée

Le domaine de l'IA a été reinvesti depuis plusieurs années après avoir connu une première heure de gloire dans les année 80. Cette première vague trouve sa naissance dans l'apparition des réseaux neurones artificiels, même si l'idée de l'intelligence peut déjà trouver ses sources chez Alain Turing (1950) avec une première réflexion sur l'intelligence d'une machine [Tur51] ou dans la première conférence de Dartmouth du domaine en 1956 [Hen17]. Dans cette première vague, on voit apparaître les premiers modèles de neurones artificiels et les termes "sciences cognitives" et "systèmes experts". L'IA rencontre aussi un premier grand succès en 1997 avec Deep Blue (système expert IBM) gagnant le tournoi d'échec contre Gary Kasparrov. La deuxième vague est essentiellement marquée, autour des

années 2010, par l'arrivée massive des données et une puissance de calcul plus importante pour le traitement de ces données. Ce nouveau contexte économique a permis de faire émerger différents domaines d'usage actuels, comme il en est question dans le rapport ministériel français sur l'intelligence artificielle en 2019 [min19] :

- Vision par ordinateur, apprentissage d'images
- La compréhension et génération du langage
- Apprentissage sur une donnée numérique, maintenance prédictive
- Usages analytiques : recherche, analyse d'informations, optimisation et coordination
- Usage sociaux : reconnaissance et génération d'émotions, interactions sociales
- Usage physique : navigation, motricité

Les moyens prévus à l'époque sur les années futures pour le développement de ces activités était considérable : budget d'un 1,5 milliard d'Euros était prévu par l'état français jusqu'en 2022 (Discours d'Emmanuel Macron après la remise du rapport de Cédric Villani, en mars 2018). De nombreux secteurs développent des projets de recherche et de développement en IA tels que la santé, les transports et la mobilité, le domaine de l'industrie (industrie 4.0), le domaine de l'énergie et le commerce de détail. Dans les éléments constitutifs de l'IA, on trouve les plateformes digitales avec les capteurs, les réseaux de communication, les infrastructures de calcul, les technologies algorithmiques (machine learning, réseaux de neurones, Deep Learning), les interfaces homme/machine. Les infrastructures de calcul concernent les technologies pour le stockage, pour le calcul intensif et des infrastructures distribuées avec le cloud.

Dans ce manuscrit, on s'intéresse plus particulièrement au secteur applicatif du transport autonome. Ce secteur est particulièrement lié à des contraintes portant sur l'embarqué qui se traduisent en contraintes de taille, poids et énergie soient SWaP (Size, Weight and Power) auxquelles on peut rajouter la contrainte de coût (SWaP-C). Le choix du développement de l'IA dans le domaine des véhicules autonomes, qui a été fait dans ce manuscrit, est essentiellement conjoncturel et dû au contexte des projets et des collaborations internationales mis en place tout au long des travaux recherche. Les diverses techniques et méthodologies établies pour les drones autonomes pourraient être éventuellement transposables à d'autres secteurs comme la santé ou l'industrie 4.0, cependant ces deux voies n'ont pas été explorées à ce jour. Si les travaux présentés se limitent au domaine des véhicules autonomes dans le milieu aérien, ils restent cependant représentatifs de la thématique l'IA embarquée'.

Le terme "IA embarquée" pointe vers des aspects contraints de l'IA notamment en termes d'infrastructure de calcul et de stockage pour des déploiements soumis à de fortes contraintes de temps. On trouve dans cette catégorie non seulement les drones aériens, terrestres et marins (de surface ou sous-marins) mais également les objets connectés (IoT : Internet of Thing). On associe très souvent le terme IA à des outils algorithmiques tels que le Machine Learning (ML) ou les réseaux de neurones profonds (Deep Learning -

DL) qui permettent, notamment, en vision, de classifier les images capturées par une caméra embarquée. Les versions embarquées existent dans de nombreux domaines comme celui de l'agriculture (reconnaissance de pommes, détection de maladie au niveau des plantes), de la sécurité des personnes (identification, suivi de personnes ou d'objets), surveillances de pollutions, de feu, de frontières ... Comme montré sur la figure [Reu+21], les architectures IA rentrent généralement dans une boucle d'interaction incluant les capteurs et le calculateur embarqué. Ces architectures doivent prendre en compte les données issues des capteurs, puis les conditionnent via des pré-traitements, afin de pouvoir être traitées par des algorithmes IA associés à différents modèles pouvant parfois interagir avec des humains.

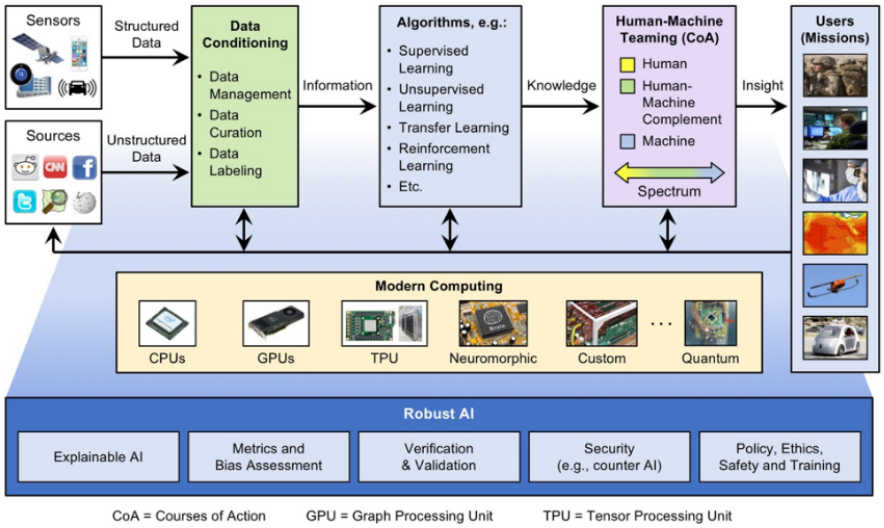

FIGURE 1.1 – Architecture IA embarquée et cycle des traitements des données (source [Reu+21])

Cependant il est restrictif de limiter l'IA aux seuls modèles des réseaux de neurones profonds, d'autres modèles plus proches des sciences cognitives peuvent être aussi des supports pour formaliser des raisonnements et prendre des décisions sans aide humaine. On peut citer comme autres modèles les réseaux Bayésiens [PR98] les forêts d'arbres, les arbres de décisions ... La figure 1.2 replace chronologiquement les différents termes classiquement utilisés dans le cadre de l'IA ; machine learning pour signifier des techniques permettant aux machines d'apprendre et Deep Learning représentant les réseaux profonds à base de réseaux de neurones.

Nous nous intéressons dans ce document à quelques modèles de l'IA, différents des réseaux de neurones profonds, pour adresser les problèmes suivants :

— la sûreté de fonctionnement du système autonome en utilisant des méthodes probabilistes graphiques de type réseaux Bayésiens
— la prise de décision d'un véhicule autonome en fonction des aléas de la mission ou

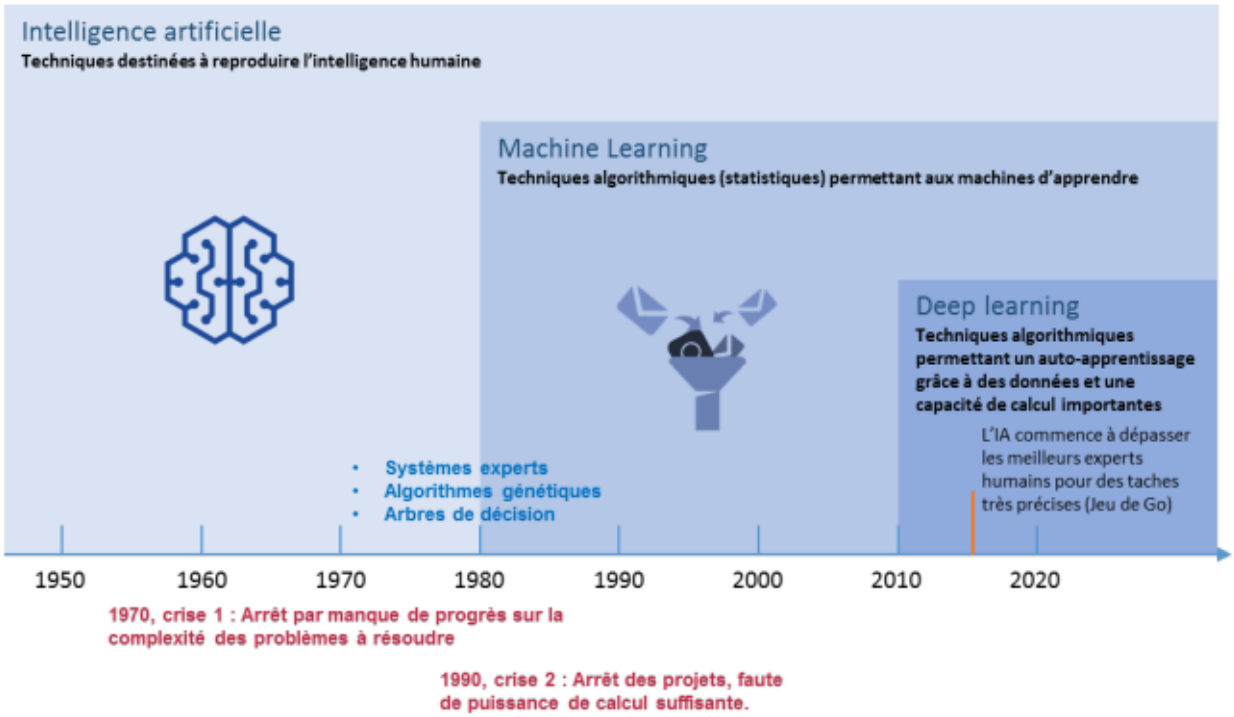

FIGURE 1.2 – IA et Machine Learning, chronologie des termes (source [min19])

intégrant des préférences humaines

— l'apprentissage en ligne pour renforcer les modèles embarqués existants

Ces thématiques sont développées plus en détail dans les chapitres 3 et 4 de ce manuscrit.

Pour mieux comprendre les enjeux des applications embarquées que l'on peut rencontrer dans le domaine des véhicules autonomes, nous abordons leur singularité dans le cadre des drones aériens autonomes (UAV) et puis la nécessité de conception d'accélérateurs matériels dédiés aux calculs embarqués sur ces systèmes.

1.1.2 Autonomie des drones aériens et complexité des systèmes

L'organisation internationale de l'aviation civile (ICAO `https://www.icao.int`) classe les avions sans pilote selon deux catégories selon la circulaire 328 AN/190 [Cir11] : les véhicules aériens pilotés à distance et les véhicules aériens autonomes (UAV pour Unmanned Aerial Vehicle ou UAS pour Unmanned Aircraft System). L'UAV peut éventuellement peut faire appel lors du décollage et de l'atterrissage à l'assistance d'un pilote à distance, mais le reste de la mission est effectué de manière complètement autonome. L'autonomie est définie comme la capacité propre d'un système à détecter, à percevoir, à analyser, à communiquer et à opérer avec les éléments de son système sans l'intervention du pilote.

Les tentatives pour définir les niveaux d'autonomie ne sont pas nouvelles. Datant des années 70, la classification de Shéridan est une des plus utilisées [SV78]. Cette classification propose dix niveaux d'autonomie allant du niveau 1 où l'humain conserve

le plein contrôle système au niveau 10 où l'ordinateur a le contrôle total. Une version révisée a été ultérieure présentée dans [PSW00]. Cette définition fondatrice a inspiré le fonctionnement de nombreuses autonomies modernes. De nos jours, la plupart des classifications d'autonomie sont basées sur l'Observe-Orient-Decide-Act (OODA) proposé par l'US Air Force. Pour les systèmes autonomes, l'US Air Force a utilisé ce cadre pour définir 11 niveaux d'autonomie [Sho07]. Des organisations telles que l'OTAN ont également proposé différents frameworks qui définissent le niveau humain d'interaction avec l'automatisation, via une politique d'autorisation et de contrôle des tâches (pilote) (PACT) [Clo02]. D'autres tentatives génériques pour définir les niveaux d'automatisation pour les UAS sont données dans [MAM19]. Le cadre ALFUS [Hua+05] est un autre outil de classification commun pour définir les niveaux d'autonomie. Récemment, une extension de ce cadre a été présenté par Kendoul [Ken12].

Pour saisir la performance d'un UAS du point de vue technique et opérationnel, le modèle de performance ALFUS peut être utilisé. Dans le cadre d'ALFUS, un niveau d'autonomie est défini en pondérant le score métrique pour trois aspects, à savoir l'indépendance humaine (HI), complexité de la mission (MC) et complexité environnementale (EC). L'évaluation HI correspond à un pourcentage d'interaction avec l'humain pendant toute la mission ou une partie de la mission, MC définit la complexité de la mission en fonction de types de tâches à effectuer au cours de la mission en prenant en compte la capacité d'adaptation en fonction du contexte, EC caractérise la complexité de l'environnement en fonction du contexte géographique (urbain ou rural), en fonction de la dynamique de changement de contexte (densité des obstacles dynamiques) et en fonction de la présence de perturbations (perturbations climatiques ou menaces de tout ordre pouvant entraîner des défaillances du système). Afin d'exprimer le degré d'autonomie d'un engin autonome, on peut ainsi établir une cartographie utilisant les métriques ALFUS MC, HI et EC avec une échelle de 1-10 comme proposé dans [Mej+21]. La valeur maximale 10 de l'autonomie lorsque le véhicule est capable d'évoluer dans le cadre d'une mission complexe (nombreuses actions à effectuer, forte adaptabilité en fonction du contexte), dans un environnement difficile (perturbé, à forte variabilité) et en totale indépendance vis à vis de l'humain (pas d'interaction humaine). Les autres niveaux d'autonomie sont précisés dans la figure 1.3. Par exemple, pour une application d'inspection d'infrastructures ou de livraison de paquets, un certain nombre de tâches peuvent être effectuées en complète autonomie comme la navigation avec un système de vision pour l'atterrissage, l'évitement d'obstacles et peuvent avoir ainsi des complexités de mission MC, et d'environnement EC assez hautes (évolution dans un environnement varié urbain/rural). De manière générale et assez intuitive, plus les métriques HI, EC et MC sont importantes (entre les valeurs 7 et 10), définissant alors un degré d'autonomie important (entre les valeurs 7 et 10), plus le nombre de tâches à effectuer sur le drone augmentent pour gérer l'absence de décision humaine et pour traiter les différents cas de figure introduits par la complexité de la mission et de l'environnement.

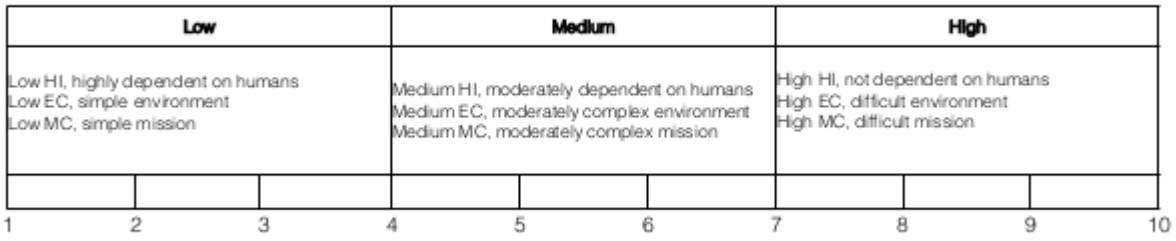

FIGURE 1.3 – Degré d'autonomie associé à la complexité de mission (MC), à l'environnement (EC) et à l'indépendance humaine HI (Human Independance)

Il est important de noter que le nombre croissant de tâches effectuées de façon autonome par les drones de nouvelle génération implique de contrôler de façon beaucoup plus rigoureuse le bon fonctionnement du système embarqué. Il est ainsi fondamental de mettre l'accent sur des fonctionnalités telles que :

1. observer correctement le système, détecter des défaillances en fonction du contexte de la mission et diagnostiquer les erreurs. Quels évènements prendre en compte ? Quelles observations sont à retenir ?
2. définir des fonctionnalités pour assurer la surveillance du système de manière systématique dans le contexte de mission en intégrant des observations de l'environnement extérieur, en plus du calculateur embarqué, de l'ensemble de capteurs et des différents composants hardware définissant l'engin autonome. Comment prendre en compte l'ensemble des aléas associés à une mission ? Quel modèle choisir pour évaluer le bon fonctionnement du système ? Comment spécifier les moniteurs de manière simple sans expertise sur le modèle sous-jacent ?
3. intégrer de manière non-intrusive ces moniteurs dans le système en faisant le choix d'une version matérielle, afin de ne pas perturber le fonctionnement du système. Comment faire la mise en oeuvre matérielle des moniteurs ?
4. mettre en place un moteur de décision intelligent intégrant les moniteurs précédents. Comment spécifier cette décision ? Quel modèle de décision ? Comment intégrer le diagnostic ?
5. mettre à jour le système en ligne à partir des informations fournies par les moniteurs et les mécanismes de décision. Comment effectuer cette mise à jour ? Quand prévoir les mises à jour ?

Ces différents questions sont traitées dans les chapitre 3 et 4. Dans le chapitre 3 nous aborderons les problèmes de sûreté de fonctionnement dans les drones autonomes en proposant des moniteurs Bayésiens embarqués. Dans le chapitre 4, les prises de décision faites dans le cadre des missions de drones autonomes sont faites en essayant d'intégrer

différents aspects : la sécurité de la mission, les préférences humaines. Les points dans ces chapitres 3 et 4 seront traités avec le souci de couvrir un maximum d'aspects allant de la spécification de haut niveau jusqu'à la réalisation sur carte embarquée, afin de mettre en place des solutions efficaces intégrant les contraintes matérielles. La perspective d'une mise en oeuvre embarquée permet de rester dans le fil directeur de ces travaux qui reste la conception d'accélérateurs matériels qui font l'objet du chapitre 2.

1.1.3 Conception d'accélérateurs matériels pour du calcul haute performance embarqué

Dans le cas des drones aériens, on peut classer les fonctions embarquées des UAS en cinq catégories : 1) vol, 2) navigation et guidage, 3) application, 4) sécurité et 5) mission. Les fonctions concernant le vol sont prises en charge par le pilote automatique, qui utilise généralement des estimateurs d'état reposant sur des lectures de capteurs, des boucles de contrôle et de stabilisation rapide, et de contrôle de l'actionneur. La navigation et le guidage reposent sur des lois d'orientation et des routines de planification de trajectoire qui définiront et maintiendront l'aéronef dans la position optimale itinéraire (généralement) en tenant compte de l'application et des priorités de la mission. Par exemple, dans cette catégorie, nous pouvons trouver des routines pour planifier et exécuter un chemin pour suivre le meilleur itinéraire en optimisant la consommation de la batterie, la qualité des données et le temps passé. L'application définit généralement les actions principales et les traitements à mener durant la mission. Ils prennent place dans des domaines tels que l'agriculture de précision, l'inspection des infrastructures, l'inspection souterraine et la livraison de colis, qui sont quelques-unes des utilisations les plus explorées par l'industrie avec des UAS. La mission traite des tâches de haut niveau dont l'UAS est responsable au-delà du vol, elle comprend des tâches autonomes telles que la planification de la mission, la surveillance du bon fonctionnement du système, la prise de décision et la gestion des ressources. Enfin, la sécurité fait référence aux tâches qu'un UAS doit exécuter pour assurer la sécurité des personnes et des biens. Cela permet également aux UAS de se conformer aux exigences de l'organisme de réglementation en matière de vol dans l'espace aérien civil comme détecter et éviter les obstacles, gérer l'espace aérien, utiliser des procédures d'urgence, détecter et isoler des défauts.

Dans chaque catégorie de fonctions, on peut définir un ensemble de tâches qui devront être exécutées sur le système embarqué. On peut évaluer grossièrement cette charge de calcul associée à l'ensemble des tâches comme envisagé dans [Mej+21]. En prenant l'exemple de la distribution de colis pour un UAV, les tâches à effectuer sont nombreuses (évitement d'obstacle, planification de chemins, détection de site pour atterrissage d'urgence, suivi de personne, encryptage de vidéo). Cette charge peut facilement dépasser la centaine de Giga opérations flottantes par seconde (GFLOPs) lorsque plusieurs applications s'exécutent

simultanément en se basant sur des applications significatives de l'état de l'art. Actuellement, ce type de performances n'est pas accessible sur les périphériques embarqués en raison des contraintes SWaP. Même si les performances de pointe récentes sur les GPUs, CPUs et FPGAs semblent prometteuses pour atteindre des TFLOPS pour les opérations de base[VN14], ces scores sont loin de ceux obtenus lorsque des applications réelles sont prises en compte. L'une des principales raisons est le mur de mémoire (memory wall, problème de bande passante entre CPU et mémoire), qui empêche l'exploitation complète du parallélisme théorique. Certaines architectures dédiées mises en oeuvre sur FPGAs peuvent répondre aux attentes de performances pour certaines applications spécifiques. Par exemple, une solution générique pour l'apprentissage de réseaux de neurones profonds est proposée dans [Mot+16] atteignant 84 GFLOPs avec un Virtex 7. Dans [Zha+18], les auteurs rapportent des performances maximales impressionnantes de 636 GFLOPs en utilisant une implémentation du Caffeine Deep Learning avec le même appareil. Ils montrent également des gains de performances et d'énergie sur un serveur Xeon à 12 cœurs de $7,3$X et $43,5$X respectivement. Malgré une performance impressionnante, les FPGAs, dans ce cas, n'exécutent qu'une seule application à la fois. L'activation des FPGAs pour une reconfiguration à la volée afin d'exécuter plusieurs applications pourrait donc être une fonctionnalité souhaitable.

Face à la complexité des applications à embarquer sur un véhicule autonome, un certain nombre de problématiques sont associées à la mise en oeuvre de ces applications :

1. explorer le parallélisme des applications pour effectuer des mises en oeuvre performantes et efficaces au niveau énergétique. Quelles techniques de parallélisation utiliser et pour quelle cible matérielle ? Que peut-on espérer gagner au niveau énergétique ?
2. faciliter la mise en place d'accélérateurs matériels pour des non-experts à partir de spécifications de haut niveau. Quel niveau de spécifications proposer pour la description de l'application afin de pouvoir s'adresser au plus grand nombre d'utilisateurs ? Si des modèles spécifiques d'IA sont utilisées (comme les réseaux Bayésiens), il est souhaitable de reformuler leurs spécificités afin de ne pas avoir à maitriser ce modèle.
3. tester et évaluer les solutions sur les cibles matérielles choisies afin de montrer l'intérêt de la mise en oeuvre choisie. Quel support matériel pour quel type d'application en considérant le contexte de la mission du drone ? Des mécanismes d'adaptation et de reconfiguration peuvent être utiles pour s'adapter aux variations et aléas de la mission ? Quel processus de reconfiguration peut-on alors mettre en oeuvre ?

Nous aborderons ces problématiques de manière générale dans le chapitre 2 et puis de manière plus ciblée avec les applications de monitoring et de décision dans les chapitres

suivants.
Dans ce document, nous nous intéressons plus spécialement aux plateformes reconfigurables pour embarquer de l'IA dans un contexte FPGA SOC afin de pouvoir mieux évaluer le bénéfice associé au support FPGA. En effet, en utilisant une plateforme de type FPGA SOC, on peut comparer une implémentation sur FPGA à une mise oeuvre logicielle au niveau de leurs performances et consommation respectives, notamment en utilisant un processeur hybride comme le processeur Zynq de chez Xilinx qui incorpore sur un même circuit un processeur ARM et une partie logique de type FPGA.

1.2 Positionnement scientifique

Dans cette section, nous nous intéressons à la thématique de l'IA embarquée au sens large. Nous analysons l'existant en termes de plateformes matérielles proposées par les fournisseurs d'accélérateurs matériels afin de montrer les tendances actuelles dans ce domaine. Puis nous observons les tendances dans les communautés scientifiques familières avec la conception d'accélérateurs avant de présenter les contributions de ce travail.

1.2.1 Les communautés industrielles et scientifiques autour de IA embarquée

Les fournisseurs d'accélérateurs pour l'IA

Dans la course à la performance pour l'IA, on peut noter en 2020 le proposition de Cerebras WSE (presentation à DAC `https://www.cerebras.net/`), avec une puce $56,7$ fois plus grosse que le plus grand GPU, possédant 400000 coeurs de calculs possédant 18 Gigabytes de memoire 'onchip' et une bande passante de 9 petaBytes par seconde. En 2022, ce sont 850000 coeurs avec une bande passant de 20 petabytes par seconde qui sont annoncées dans les nouvelles versions de ce type de circuit.

Dans le cadre du Deep Learning, les fournisseurs de FPGA ou GPU proposent des circuits spécifiques comme l'Alvéo (de Xilinx), MPPA (de KalRay). Des frameworks de conception sont aussi proposés comme Tensorflow (`https://www.tensorflow.org/`), Pytorch (`https://pytorch.org`), Caffe (`https://caffe.berkeleyvision.org/`). Ils s'inscrivent dans une course aux performances qui se conjugue avec l'optimisation énergétique. Dans la figure 1.4, sont positionnés quelques uns des circuits ou systèmes développés pour traiter des problèmes d'IA (problèmes d'inférence ou d'apprentissage) en précisant leur consommation énergétique (en Watts) et leur performance exprimée en GOPs (Giga operations par seconde). On peut remarquer que l'on peut trouver quelques super-calculateurs qui peuvent proposer jusqu'à 10 TeraOPs/W d'efficacité énergétique. Cette figure donne une idée des tendances actuelles en terme de super calculateurs mais ceux-ci ne respectent pas toujours les contraintes SWaP pour pouvoir être embarqué.

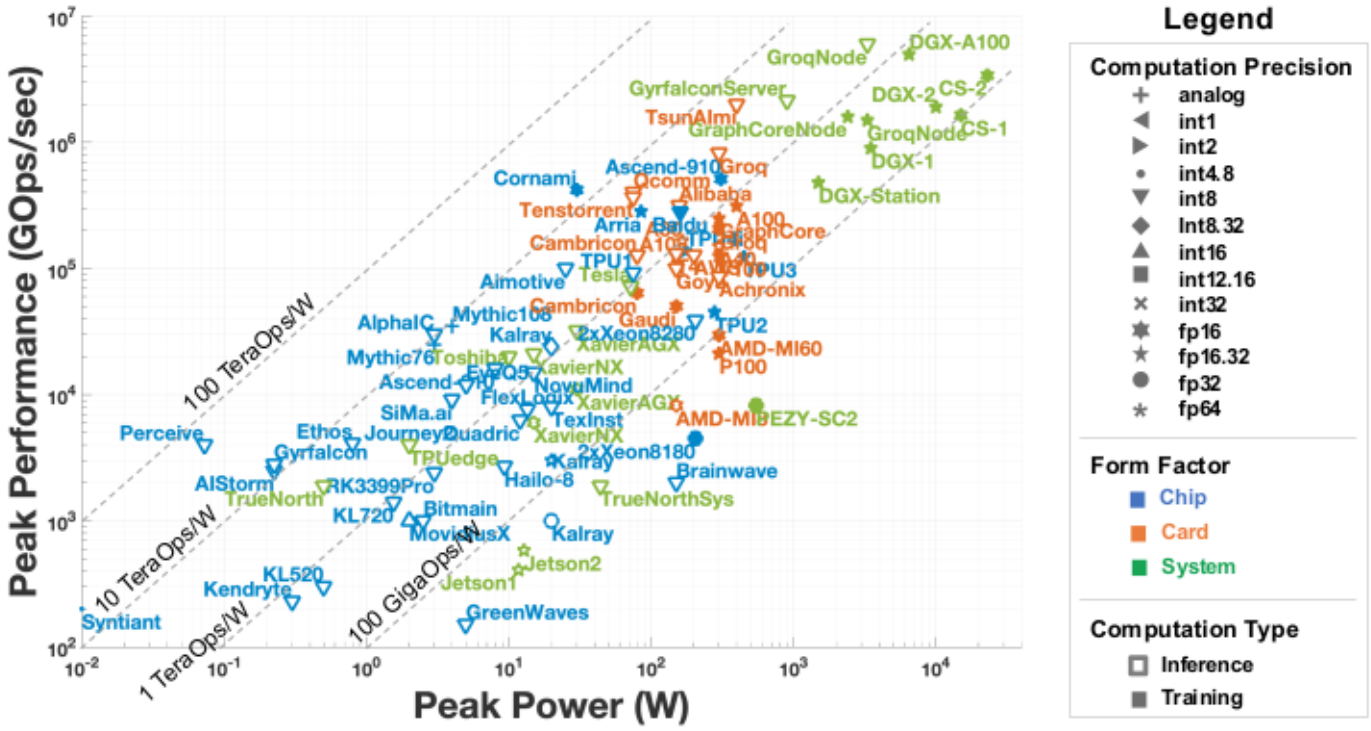

FIGURE 1.4 – Performance de crête vs. puissance dans l'annonce des accélérateurs et processeurs supercalculateur pour l'IA (source Albert Reuther [Reu+21])

Des accélérateurs matériels pour l'IA apparaissent dans différents domaines d'application de l'embarqué : Edge computing, engins autonomes comme la voiture, applications mobiles avec les smartphones, et calcul haute performance pour les robots. La faune des accélerateurs matériels existant est donnée par une étude de Yole-development faite en 2020 et illustrée dans la figure 1.5. Cette étude donne un ordre de grandeur de la consommation de ces accélérateurs pour les machines autonomes, qui est de quelques Watts à quelques dizaines de Watts. Cette information montre que ces consommations ne sont pas négligeables et peuvent conditionner fortement les temps de mission notamment pour les drones aériens.

L'IA et les communautés scientifiques des accélérateurs matériels

Le domaine des accélérateurs matériels présente un intérêt scientifique certain dans la course aux performances et mobilise de très nombreux chercheurs. Un certain nombre d'équipes de recherche sur le territoire français aborde ainsi le thème des accélérateurs matériels pour l'IA. On peut compter parmi elles, notamment les équipes CAIRN (IRISA), TRACE (IRIT), STR (LS2N), SOC (LIRMM), SEAS (CRISTAL), AlSOC / SIEL (LIP6), ARCHI (TIMA) ou le département DACLE commun au CEA LIST/LETI. On trouve aussi des équipes plus spécialisées sur les architectures neuromorphiques comme CELL (ETIS), MCSOC (LEAT). Le GDR SOC2 (groupement de recherche pour les systèmes

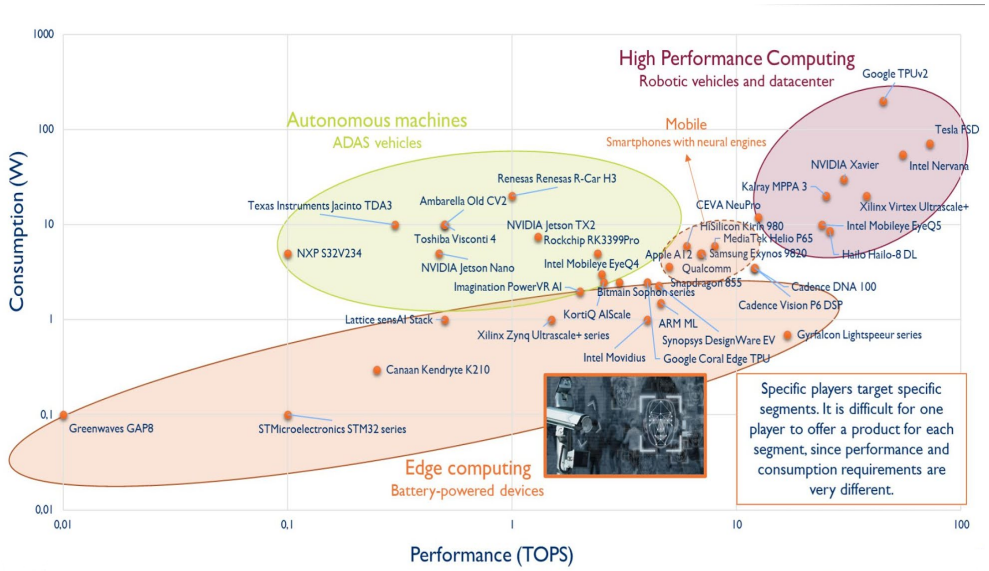

FIGURE 1.5 – Solution matérielle pour les différents domaines de l'IA (source [Dev])

embarqués mettant l'accent sur les architectures matérielles) propose depuis 2019-2020 une activité autour de l'IA et de l'embarqué. Actuellement une cinquantaine de laboratoires de recherche français participent à ce GDR et abordent certaines thématiques (outillage pour la conception, architectures dédiées et adaptation, sécurité des systèmes,...) associées à cette activité (`https://www.gdr-soc.cnrs.fr/laboratoires/`). Cette tendance à adhérer ce type d'activités autour IA embarquée devrait se renforcer avec les incitations actuelles du gouvernement français promettant 740 millions d'euros dans le cadre de la Stratégie d'Accélération Intelligence Artificielle via le Plan France 2030 [1] .

A l'international, la communauté HiPEAC (High Performance Embedded Architecture and Compilation `https://www.hipeac.net`) qui a pour mission de promouvoir la recherche et les développements en architecture des ordinateurs et en systèmes en Europe, porte aussi une attention toute particulière pour le domaine de l'IA.

1.2.2 Contributions

Mes activités de recherche s'inscrivent dans la conception d'accélérateurs matériels pour système autonome intelligent et plus généralement dans le cadre de l'IA embarquée.

Les contributions de ce travail sont déclinées selon 3 axes qui correspondent respectivement aux focus suivants :

— **Axe 1 :** Méthodes et outils pour la mise en place d'accélérateurs matériels
— **Axe 2 :** Prise en charge de la sûreté de fonctionnement dans le contexte des engins autonomes reposant sur des modèles probabilistes.
— **Axe 3 :** Mise en place des mécanismes de décision dans le contexte de planification de missions

1. `https://www.entreprises.gouv.fr/fr/strategie-d-acceleration-intelligence-artificielle`

Les document s'organise autour ces trois axes de recherches. Les différents axes (**Axe 1**,**Axe 2** et **Axe 3**) sont développés respectivement dans les chapitres 2 (*Conception d'accélérateurs matériels*), 3 (*Diagnostic et synthèse de modèles probabilistes*) et 4 (*Décision et apprentissage embarqués pour la planification de mission*).

La liste des publications et productions scientifiques associées à chacun des Axes de recherche sont détaillées dans le document Annexe (Parcours professionnel).

Contributions pour l'Axe 1 (Conception d'accélérateurs matériels)

Dans cette thématique qui correspond à l'Axe 1 de mes travaux, je me suis intéressée aux méthodes et outils pour la conception d'accélérateurs matériels qui s'est faite dans deux contextes scientifiques différents pour les projets de recherche abordés :

1. Dans un premier temps et au démarrage de mon activité de recherche à l'UBO, les outils de HLS (High Level Synthesis) disponibles étaient essentiellement des outils académiques. J'ai fait alors le choix, pour cette première période, de participer le projet collectif Madeo, porté à l'époque par l'équipe Architecture et Systèmes de l'UBO. J'ai travaillé sur ces axes de recherche dans ce contexte jusqu'en 2011.
2. Dans un deuxième temps, et avec l'émergence d'outils de HLS commerciaux, j'ai envisagé une approche plus intégrée des propositions concernant la génération d'accélérateurs matériels.

Durant la première période, je me suis intéressée à des problèmes "de niches" qui sont les suivants :

— prise en charge par l'outil d'une arithmétique non standard comme l'arithmétique dans les corps de Galois, en proposant une spécification de haut niveau et des techniques de compilation efficace pour un mapping sur FPGA. Cet outillage a permis de modéliser et synthétiser un turbodécodeur pour le projet ValMadeo,
— modélisation des supports technologiques non standards dans le but de pouvoir déployer facilement une application sur un support de type crossbar. Pour ce type de projet, l'ingénierie des modèles a permis d'envisager une adaptation de l'outillage pour différentes architectures nanométriques.

Durant la deuxième période, les outils/méthodologies de recherche développés pour les mises en oeuvre matérielles se sont appuyés sur des outils commerciaux de haut niveau existants. Dans cette approche intégrée, on s'est intéressé à certains aspects avancés qui viennent compléter les chaines de conception existantes :

— la gestion du partitionnement logiciel-matériel pour les applications embarquées contraintes en performance et en ressources
— la gestion de la reconfiguration dynamique partielle pour les supports de type FPGA

— l'aide à la conception IP spécifiques pour des familles d'application et notamment dans le cadre de l'IA embarquée

En révisant la stratégie de conception et en proposant des interfaces vers des outils de HLS commerciaux, j'ai pu bénéficier d'un ensemble d'outillage pour l'estimation, la validation des accélérateurs matériels déjà en place dans le cadre de développements industriels. Dans cette approche intégrée, la conception d'accélérateurs sur nanotechnologies n'a pas pu être poursuivie faute d'outils back-end disponibles.

La thématique concernant les méthodologies de conception d'accélérateurs matériels est un fil rouge pour l'ensemble du manuscrit et se trouve aussi développée dans les chapitres 3 et 4 qui sont plus particulièrement dédiés au contexte de l'IA embarqué.

Contributions pour l'Axe 2 (Diagnostic et synthèse de modèles probabilistes)

Dans la définition de moniteurs capables de renseigner sur les aléas de la mission, il est important de pouvoir capturer des informations correspondant au contexte de la mission. Les capteurs installés à bord de l'UAV permettent de donner quelques informations sur l'environnement dans lequel évolue le drone, les moniteurs internes complètent les informations relatives aux défaillances internes. En faisant co-habiter modèle interne de défaillance et contexte d'apparition relevé par un capteur, on peut dresser un état de santé du système utilisant ces deux types d'information. Nous avons choisi les réseaux Bayésiens pour aider à formaliser la définition de cet état de santé. Plusieurs contributions concernent cet axe. La première est la mise en place d'une proposition pour caractériser de manière automatique ces états de santé en se basant sur le formalisme FMEA (Failure Mode and Effects Analysis) et en générant les réseaux Bayésiens correspondants. La deuxième contribution est une proposition de génération de code pour définir une implantation sur SoC hybride (FPGA SoC). De façon plus précise, nous avons dans cet Axe 2 travaillé sur les points suivants :

— Définition des états de santé avec des réseaux Bayésiens. Une spécification simplifiée à partir d'une formulation de type FMEA est proposée à l'utilisateur afin de simplifier la définition de ces éléments [Zer+17a].
— Synthèses sur support hybride de moniteurs intelligents embarqués. Ce volet propose une interface (pré-traitement) avec des outils de HLS commerciaux pour une implantation sur plateforme de type FPGA-SoC. Nous nous intéressons à la mise en oeuvre embarquée des moniteurs probabilistes associés aux états de santé qui a été abordée dans le cadre de la thèse de Sara Zermani [Zer+15b] [Zer+15a].

Contributions pour l'Axe 3 (Décision et apprentissage embarqués pour la planification de mission)

La planification de mission est un problème nécessitant une réflexion au niveau des modèles à prendre en compte. Parmi les approches possibles, nous nous sommes concentrés sur trois propositions : diagramme d'influence, MDP (Markov Decision Process) et préférences humaines. Le premier travail consiste à mettre en place tous les éléments du modèle permettant de formuler la planification de mission. Le deuxième travail vise à raffiner les différents paramètres du modèle choisi. Ce point correspond à la phase d'apprentissage. Un troisième volet concerne les versions embarquées des formalisations proposées. Pour définir des versions embarquées des mécanismes de décision, nous avons défini de nouvelles méthodes de compilation et proposé des mécanismes de reconfiguration (éventuellement dynamique) pour une mise en oeuvre effective dans le cadre de planification de mission. Dans cet Axe 3, les différentes contributions sont listées ci-dessous :

— Décision pour le mission planning avec des diagrammes d'influence [ZDE17b][ZD19], avec des MPDs[Hir+18b][Hir+18c][DZH20] ou modèles intégrant des préférences humaines [Kha+18c]
— Apprentissage et raffinement des modèles : pour les modèles à base de réseaux Bayésiens [HDD17], et pour les modèles à base de préférences humaines [Kha+18b][Kha+18a][Kha+19]
— Décision/Apprentissage embarqués avec de nouvelles méthodes de compilation[Hir+18e], et et d'adaptation par reconfiguration (reconfiguration dynamique) [Maz+19][Maz+20]

1.2.3 Motivations

L'ensemble des travaux réalisés dans les trois axes précédents et présentés dans ce manuscrit s'inscrit dans une logique générale qui peut se résumer selon les objectifs génériques suivants :

1. Faciliter les mises en oeuvre parallèles d'applications de type 'computer bound' (limités pour leur volume de calcul) en proposant des méthodes et outils de type HLS avec une programmation à haut niveau.
2. Promouvoir les architectures parallèles et reconfigurables. Des architectures systoliques dédiées aux architectures hybrides associant FPGA, CPU et GPU dans un seul circuit. Utilisation de ces architectures dans un contexte embarqué en prenant en compte des contraintes énergétiques.
3. Anticiper les défaillances de l'application et du système en intégrant dans les accélérateurs matériels des modèles probabilistes permettant de gérer les anomalies.
4. Faciliter les prises des décisions dans le cadre de véhicules autonomes et guider leur déploiement sur les systèmes embarqués.

Chapitre 2

Conception d'accélérateurs matériels

Ce premier axe concerne la génération d'un accélérateur matériel à partir d'une spécification de haut niveau. Il regroupe les activités autour de la synthèse de haut-niveau utilisant deux approches méthodologiques très différentes : 1) la première se concentre sur la définition d'un nouveau concept d'outil qui a été en rupture avec les outils conventionnels existants, 2) la deuxième propose une extension des outils commerciaux existants pour faciliter la prise en compte des innovations proposées. Dans la première approche ont été abordés des problèmes de synthèse non conventionnelle intégrant des types spécifiques (ex type sur corps de Galois), et des problèmes de modélisation des supports nanométriques en utilisant des outils académiques mis en place dans le projet Madeo. La deuxième période a commencé avec l'arrivée sur le marché d'outils HLS performants comme $Vivado\_HLS$ de $Xilinx$. Cette nouvelle approche a permis la mise en place efficace de nouvelles méthodologies autour de la reconfiguration dynamique et de la prise en compte des applications IA dans l'embarqué.

2.1 Contexte des travaux

A partir des années 90, des outils de HLS voient le jour dans le monde académique. Ma thèse 'Géneration automatique de circuits avec Alpha du Centaur' s'inscrit dans cette vague en 1993. En arrivant à l'UBO au département informatique, j'ai continué sur ce registre via divers projets. Les premiers travaux ont consisté à utiliser des approches "objet" pour simplifier le processus de synthèse et à le rendre portable sur différents supports reconfigurables. Dans ce cadre, des applications de type turbo-décodeur ont été plus particulièrement étudiées en proposant des méthodes de synthèse spécifiques pour les calculs dans les corps finis. Un deuxième type de travaux a porté, toujours en utilisant le cadre de cet outil, sur la modélisation des familles architectures associées à des nanotechnologies émergentes dans les années 2000. L'approche utilisée dans ces travaux était une approche dirigée par les modèles mettant en avant des modèles génériques adaptés aux architectures crossbar nanométriques. Quelques années plus tard, ces outils académiques ont été délaissés au profil d'outils commerciaux comme Vivado-HLS de Xilinx qui proposent des mises en oeuvre sur FPGA à partir de spécifications haut niveau en C/C++/systemC. Les outils/méthodes sur lesquels j'ai travaillé par la suite, proposent des pré-traitements pour faciliter la compatibilité avec ces outils commerciaux. Les activités

concernent le déploiement d'applications embarquées sur support reconfigurable hybride dans le cadre de la vision et des applications Radar sont développées dans ce chapitre. La mise en oeuvre des applications développées dans le cadre du diagnostic, décision et apprentissage utilise également cette dernière approche et est abordée de manière plus ciblée dans les Axes 2 et 3 de ce rapport.

2.2 Conception d'accélérateurs matériels : approche dédiée ou intégrée

La conception d'accélérateurs matériels repose sur un ensemble de tâches permettant de transformer une description de haut niveau en une mise en oeuvre exécutable sur un support matériel ciblé. Les différentes étapes de transformation permettent d'ordonnancer les différentes tâches de l'application et de leur allouer des ressources matérielles et en prenant en compte différents critères tels que ressources, performance, énergie pour optimiser les mises en oeuvres ciblées. Avec l'évolution des technologies vers les dimensions nanométriques et la considération de contextes de fonctionnement éventuellement perturbés pouvant générer ou transmettre de données erronées, l'ensemble précédent de critères a été étendu à la sécurité de fonctionnement et sa capacité à être résilient aux défauts et aux attaques. La mise en place des outils de conception de ces accélérateurs repose sur les questions suivantes relatives aux outils de conception disponibles pour la communauté scientifique :

- les spécifications de l'application peuvent-elles effectuées à haut-niveau, c'est à dire à un niveau proche d'une spécification fonctionnelle exécutable ?
- les transformations proposées sont-elles capables de prendre en compte correctement les spécificités de l'architecture en intégrant les contraintes de parallélisme, de ressources, d'interface, de mémoire mais aussi leur capacité de résilience ?
- les outils de mise en place sont-ils faciles d'utilisation, sont-ils efficaces pour l'application visée ? Offrent-ils une bonne adéquation Algorithme-Architecture-Technologie ?

Les réponses à ces questions sont très fortement liées aux hypothèses de travail. Deux hypothèses de travail ont été ici envisagées :

1. dans une première période, incluant mes travaux de thèse et la première partie de mes travaux de recherche au sein de l'UBO, le choix scientifique a été de proposer des outils et des méthodes innovantes en totale rupture avec les outils existants à cette époque.
2. dans une deuxième période et en raison de l'apparition d'outils de HLS sur le marché, les aspects novateurs concernant la conception d'accélérateurs ont été pensés en vue de leur intégration dans les chaînes de conception existantes.

La section 2.3 présente les motivations et contributions des outils dédiés en rupture

développés dans la première période de recherche. La section 2.4 sur les outils intégrés présente la philosophie actuelle pour mes recherches concernant le développement des outils d'aide à la conception d'accélérateurs matériels.

2.3 Méthodes de synthèse avec des outils dédiés en rupture

Ce volet de recherches correspond à des activités menées dans la première période d'activités scientifiques. Dans le cadre de l'outil Madeo, j'ai contribué plus particulièrement aux points suivants :

1. synthèses symboliques avec ValMadeo, proposition de méthode et outil à partir de spécifications de haut niveau (code Smalltalk)[DLP99][Lag+02] [And+05a]. Cette thématique a été développée dans le cadre du projet ValMadeo pour proposer une méthode permettant des mises en oeuvre efficace d'opérations de décodage de turbocodes sur FPGA[Dez+06c][Dez+07][Gou+08].
2. synthèses pour support de type crossbar dans le cadre des nanotechnologies à base de nano-fils de silicium. De nouveaux modèles architecturaux de type crossbar ont été pris en compte dans le processus de synthèse au niveau de l'outil Madeo[Dez+09]. Une adaptation aux crossbars développés par A. Moritz à l'UMASS a été proposée [Mor+07]. Cette thématique a été plus particulièrement abordée dans la thèse de Ciprian Téodorov en coopération avec UMass [Teo+11].

2.3.1 L'approche objet et ingénierie de développement

Pourquoi toujours réinventer la roue à différentes étapes de la chaîne de conception ? Cependant cela est souvent le cas, les chaînes de conception de circuits sont souvent dédiées à la technologie et au fournisseur (Xilinx, Altera) et les flots de conception sont complètement disjoints et ne permettent pas de passer facilement d'une technologie vers une autre, même au sein d'une même gamme de produits d'un même fournisseur. Cependant, même si ces chaines de conception ont des cibles matérielles différentes, elles peuvent utiliser des modèles communs (graphe de représentation interne des spécifications) et des algorithmes d'optimisation communs. Beaucoup d'efforts pourraient être factorisés dans la mise en place d'outils d'aide à la conception d'applications pour ce type d'architecture. Comment s'affranchir des différences technologiques en proposant des outils adaptables, paramétrables tout en ne perdant pas en efficacité ? C'est le défi ambitieux que nous avons souhaité aborder dans cette première période avec le projet Madeo.
Pour relever ce défi, l'approche objet et l'ingénierie dirigée par les modèles ont été choisies. Cette approche a permis la mise en place de solutions génériques et adaptables. L'approche

objet, par rapport à d'autres approches plus conventionnelles, a permis de réduire la complexité des problèmes d'adaptation, notamment sur deux points :

— l'encapsulation des données et des comportements à travers des classes abstraites permet de factoriser un certain nombre d'éléments et permet de construire des spécificités d'implémentation avec des classes concrètes

— le typage dynamique de certains langages comme Smalltalk a pour vertu de considérer le calcul sans a priori sur la mise en oeuvre des type de données facilitant ainsi la généricité des calculs modélisés.

Je me suis intéressée à ces deux points dans le cadre de la synthèse symbolique sur corps de Galois pour la mise en oeuvre de turbo-décodeurs et de la conception de framework pour les nanotechnologies de type crossbar. Ces deux volets sont développés dans les sections suivantes.

2.3.2 Synthèse symbolique d'arithmétiques non conventionnelles

Le but de cette étude a été de faciliter la prise en compte d'arithmétique non standard en opérant sur des données/calculs symboliques. Le typage des données est une contrainte qui fige très rapidement les optimisations possibles et peut amener des biais dans les calculs. L'idée de cette étude était de s'affranchir d'une mise en oeuvre trop précoce qui pourrait entraîner des calculs inutiles. Les travaux autour de la synthèse symbolique ont permis d'aborder le calculs sur les corps de Galois de manière originale. On modélise via un type spécifique les propriétés des opérations s'effectuant dans ce cas d'étude. Le cadre objet permet une mise en oeuvre simple et efficace. Une fois les calculs effectués et les opérations simplifiées, on opère un premier ciblage sur une architecture de type FPGA en découpant en calculs élémentaires sur des tables de 'Look Up' abstraites. Le typage des données se fait après ce découpage permettant de limiter les mises en oeuvre au strict nécessaire. Cela signifie que l'on peut tailler le type de données en fonction des calculs effectués afin d'éviter de propager des espaces de valeurs sur-dimensionnés.

Le choix du typage dans les mises en oeuvre matérielles est une source d'optimisation. Les approches classiques se bornent, dans le cas des données réelles, au choix entre type flottant ou virgule fixe sur une largeur de bits de données l'on essaie d'estimer. Ce choix de la largeur des données est capital et reste une option d'exploration autant dans les domaines du traitement signal/image que dans les domaines de l'IA embarquée avec, par exemple, les réseaux de neurones pour les classifications [GMG16]. Il est important d'avoir cependant à l'esprit que les habitudes de programmation peuvent générer des complexités pour la mise en oeuvre optimisée. Le passage par des classes génériques représentant le typage (comme en C++, ou systemC) est un élément de réponse mais cette proposition ne permet que des adaptations à la marge (choix de la largeur des champs partie entière, partie fractionnaire, ou choix de la largeur des champs mantisse et exposant) des normes

IEEE classiques pour définir par exemple des réels. Un autre élément de réponse concerne la révision du complet choix du type pouvant s'appuyer sur une multitude de standards ou sur de nouvelles représentations. Une réponse possible se trouve aussi dans le choix d'une arithmétique multi-précision comme BigNum [St 06]. BigNum donne une solution pour s'affranchir des problèmes de précision engendrés par les types classiques proposés sur les réels. Les intervalles creux de valeurs pour les entiers peuvent être aussi une source d'optimisation difficile à prendre en compte dans un contexte standard mais pourrait l'être plus facilement si l'on dispose d'une structure de données adaptée et de nouveau modèle d'arithmétique pour coder les opérations sur ce type de données. Ce problème devient un problème de représentation des nombres pour générer des opérateurs efficaces pour les opérations souhaitées et pour les mises en oeuvre matérielles. Même si l'encodage des données repose généralement sur des solutions classiques de réels en virgule flottante ou à virgule fixe qui sont généralement utilisées dans les outils de conception d'accélérateurs, on peut cependant noter quelques exemples de librairies utilisant des représentations différentes comme par exemple sur le système à base logarithmique [DD07] pour aider à la mise en place des opérateurs arithmétiques.

Ces travaux ont été abordés avec le stage de master Caaliph Andriamisaini en 2004 sur la synthèse abstraite d'éléments de turbo decodeur en bloc pour un circuit reconfigurable[And+05b][And+05a], puis repris pendant le projet Valmadeo en collaboration avec Thierry Goubier [Gou+08].

2.3.3 Framework pour les nanotechnologies de type crossbar

La montée en puissance des nouvelles technologies dans les années 2000 (émergence de la technologie graphène et du calcul quantique) ouvre le champ à de nouvelles mises en oeuvre matérielle. Lorsque ces nouvelles technologies ont été suffisamment matures, des prototypes pour de calculateur ont été parfois élaborés. Cependant, l'utilisation de ces technologies à grande échelle nécessite l'usage d'outils d'aide à la conception. Ces outils nécessitent des années de développement avant d'être disponibles. Comment récupérer l'expertise acquise pour les technologies plus standards pour la transférer vers ces nouveaux outils ? C'était le défi mené dans la thèse de Ciprian Téodorov concernant les architectures de types crossbar [TDL08] [Teo+11]. Cette thèse entre dans le projet de collaboration avec l'université de Amherst (Massachusetts). Dans cette thèse, un certain nombre d'architectures innovantes comme NanoPLA, CMOL, FPNI, Nasic ont été étudiées. Elles utilisent une variété de composants comme des transistors à effets de champs, les composants à base de graphène, des diodes et des commutateurs moléculaires qui sont généralement interfacés à de la technologie classique à base de transistors CMOS (ou ses variantes) pour assurer notamment des interfaces cohérentes avec les autres composants standards du systèmes ou de la carte.

L'approche objet a permis de dégager des modèles communs afin de construire une approche de synthèse physique incluant les problématiques de placement/routage adaptées aux technologies à l'étude et notamment de type crossbar. Cette approche correspond à un travail de méta-modélisation des architectures introduisant des nano-composants. Elle a permis de construire un flot de conception/synthèse pour ces nouvelles technologies sur la base de réutilisation et l'adaptation d'algorithmes de placement/routage reployés dans le cadre du méta-modèle. Les travaux effectuées dans ce domaine ont montré sur un cas d'étude fourni par A. Moritz à Amherst (UMASS) que cette méthode est envisageable et qu'elle évite de définir des outils ad hoc non adaptables aux variations technologiques. La Figure 2.1 illustre ce cas d'étude d'implémentation sur architecture NASIC.

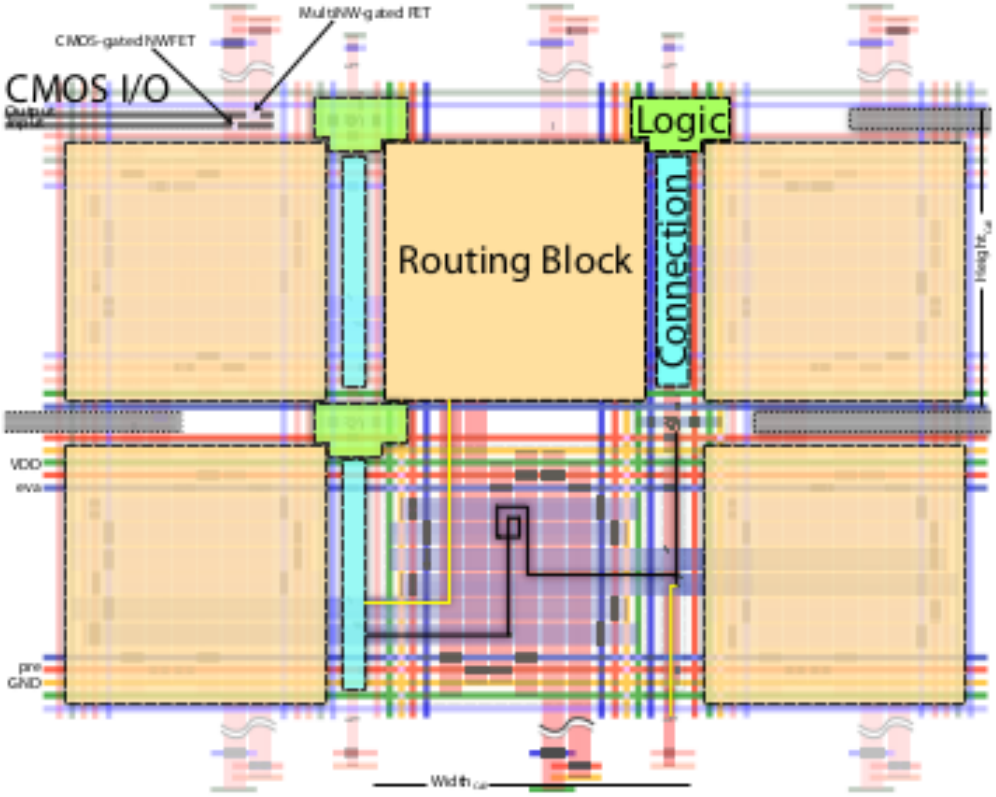

FIGURE 2.1 – Illustration d'une implémentation sur nano-technologie de type NASIC à base de nanofils de silicium (source [Teo11])

2.3.4 Conclusion sur les outils dédiés et l'approche objet

Il était très ambitieux de vouloir concevoir des outils dédiés avec des coeurs d'algorithme ou de méthodologies réutilisables. On a pu montrer que cette approche était possible et ceci avec très peu de moyens humains. Cependant, même si les résultats ont aidé à montrer les avantages de cette approche, elle reste peu viable dans le long terme pour les raisons suivantes :

— L'approche holistique des outils demande un gros effort d'ingénierie pour leur mise en place et un effort constant pour leur mise à jour. Cet effort est souvent peu gratifiant au niveau de notre communauté scientifique et très peu valorisable dans des communications scientifiques.
— Il est nécessaire de constamment prendre en compte des spécificités des évolutions technologiques et architecturales des fournisseurs de FPGA, souvent peu enclins à livrer les détails de leurs technologies.
— Le travail collaboratif est délicat lors de grosses évolutions du coeur logiciel, demandant une très bonne cohésion entre les différents développeurs et utilisateurs.

Cette approche a été très riche en enseignement et nous pouvons en retenir les aspects positifs suivants :

— on peut simplifier et réduire à des coeurs d'algorithme pour investir de nouveaux domaines. On peut ainsi simplifier l'approche méthodologique en proposant des adaptations de ces coeurs algorithmiques.
— les explorations de tout genre peuvent être effectuées et réutilisées à différents niveaux. Souvent un algorithme ILP peut être utilisé dans plusieurs contextes (synthèse de haut niveau, problème de routage, ...). Les adaptations contextuelles peuvent être facilitées en utilisant l'approche objet et ingénierie des modèles.

Je reste persuadée que cette approche conserve de grandes perspectives pour faciliter les explorations futures. Cependant, l'effort d'ingénierie reste énorme et sans support concernant ce point, cette activité reste parfois difficilement compatible avec une activité de recherche soutenue.
Cette prise de conscience m'a permis de me focaliser sur des problématiques de recherche pouvant être plus facilement compatibles avec des outils de synthèse de haut niveau efficaces et commerciaux qui ont émergé dans les années 2012-15. Les outils de synthèse de haut niveau commerciaux comme $Vivado - HLS$ de Xilinx ont été suffisamment matures pour envisager la conception d'accélérateurs autrement, en proposant des front-end compatibles avec ces mêmes outils afin de concevoir des accélérateurs adossés aux technologies actuelles ciblant essentiellement la technologie CMOS.

2.4 Approche intégrée

Dans cette section sont abordées les méthodologies de conception d'accélérateurs matériels qui peuvent s'intégrer à des plateformes commerciales existantes. Dans ce type d'approche, il est essentiel de repérer les orientations de recherche qui peuvent être apportées une plus value significative tout en pouvant cohabiter avec des méthodologies classiques de conception d'accélérateurs proposées par les fournisseurs de FPGA. Les points abordés dans ce cadre concernent :

1. Approche co-design. Ce type d'approche s'intéresse aux synthèses d'architectures

sur support hybride pour définir un système adaptatif pouvant être embarqué. Cette technique de co-design tire avantage des supports de type FPGA-SoC avec possibilités de implémentation sur FPGA, CPU et GPU. Ces techniques de co-design sont aussi abordées afin de trouver le meilleur compromis performance/énergie dans l'implantation sur SoC. Cette thématique a été abordée à avec les premiers travaux sur le co-design effectués par la post-doctorante Hanen Chenini sur le projet PICS SWARMS pour du traitement d'images embarqué[Che+15b].

2. Intégration de la reconfiguration partielle dynamique dans le flot de conception. La reconfiguration du support offre un cadre pertinent pour la mise en oeuvre d'architecture efficace. Ce point a été abordé dans les domaines d'application du traitement d'images dans le cadre du projet PICS [Che+15b] et dans le domaine du traitement Radar dans le cadre du projet RECONFIG[Maz+19][Maz+20]. Le déploiement de ces applications images ou radar s'est fait sur une carte hybride à base de FPGA dans une approche co-design proposant une coopération en logiciel et matériel. La reconfiguration dynamique peut permettre une adaptation réactive et favoriser une prise de décision en ligne d'une configuration appropriée.
3. Déploiement d'IP (Intellectual Property ou composant en général dédié à une application) pour des applications d'IA spécifiques sur cartes hybrides de type FPGA-SoC. Dans le cadre du projet VISEMAR [Le +20], une méthodologie avec une approche intégrée a été explorée pour effectuer de la classification d'images utilisant un modèle de type Deep learning. Ces travaux reposent sur l'adaptation d'ateliers logiciels qui utilisent des formats d'échange de modèles de type machine learning comme Onnx (Open Neural Network Exchange).

Ces points ont été principalement abordés dans le cadre de déploiement d'applications sur véhicules autonomes mais ils peuvent être retenus pour d'autres domaines d'applications. Dans les sections qui suivent, les grandes lignes de cette approche intégrée sont données et illustrées dans deux cas de figures : a) cas de l'intégration de la reconfiguration partielle et b) cas de la génération spécifiques d'IP à base de CNN pour la classification d'images.

Dans chapitres suivants (chapitre 3 et 4), les aspects méthodologiques dédiés au diagnostic et à la prise de décision dans le cadre des véhicules autonomes seront plus particulièrement abordés.

2.4.1 Propositions de front-end pour l'élaboration de nouveaux outils de conception

Dans l'approche intégrée, les nouvelles propositions d'outils de conception reposent sur une intégration possible avec les outils de HLS commerciaux. Cet approche permet d'adapter et d'enrichir les outils front-end existants associés à la chaîne de conception proposé par les fournisseurs de FPGA (Xilinx ou Altera). Cette chaine présente classiquement deux

étapes majeures qui sont celles associées au front-end puis celles associées au back-end comme illustré dans la Figure 2.2.

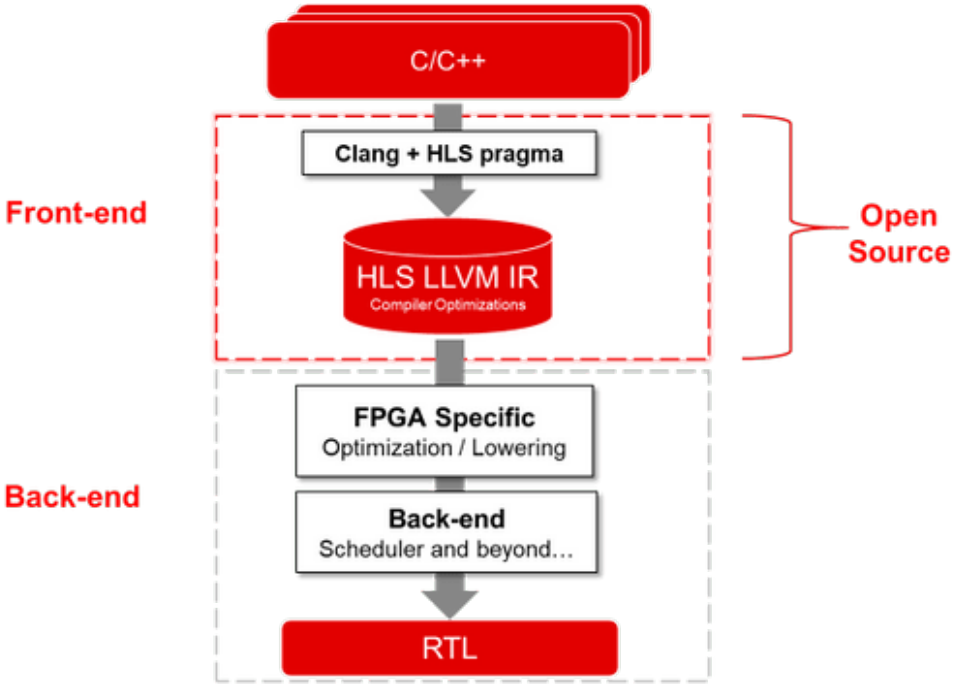

FIGURE 2.2 – Flot de conception à partir de front-end (source *forums.xilinx.com*)

La partie 'front-end' permet de passer de spécifications exécutables pouvant être décrites en C/C++/systemC et qui sont synthétisables via des outils de HLS. La partie 'back-end' représente l'ensemble des logiciels qui permettent un placement et routage effectif de l'application sur le support matériel choisi à partir d'une description RTL (généralement décrivant l'architecture sous format VHDL ou Verilog). Cette étape est très dépendante de la cible et produit in fine un bitstream qui permet de configurer le FPGA.

L'enrichissement de cette chaîne d'outils est possible dans la partie 'front-end' en proposant des extensions des outils de HLS. Actuellement, ces outils de HLS ont les caractéristiques suivantes :

— en entrée : description d'une application synthétisable exprimée avec un langage de haut niveau (en C/C++/systemC) avec quelques restrictions concernant notamment l'allocation dynamique et l'utilisation de la récursivité
— en option : des transformations de haut niveau sont proposées pour diriger l'allocation et l'ordonnancement des ressources nécessaires à la mise en oeuvre. L'usage de directives sous forme de pragmas dans le code est proposé pour gérer le parallélisme de boucle, le découpage fonctionnel, le partitionnement des tableaux, la gestion des modes d'interface pour acheminer les données.
— en validation : des outils de validation, et d'estimation concernant quelques métriques comme l'occupation des ressources matérielles, les performances, la consommation.
— en sortie : description de type RTL jouant le point d'entrée des outils de 'back-end'.

L'existence d'outils de HLS commerciaux permet une validation simplifiée de nouveaux concepts d'outils front-end pour la mise en place d'accélérateurs performants. En effet, de nombreuses extensions sont actuellement possibles. Elles permettent d'introduire des innovations à différents niveaux, notamment au niveau système et architectural.

Au niveau système : les approches co-design donnent une vue unifiée de la partie logicielle et matérielle d'un support hybride pouvant intégrer un processeur généraliste avec un FPGA ou autre macro-composant comme des GPU. Les principales questions à ce niveau concernent le partitionnement du système et le placement des différentes fonctionnalités associées à l'application sur des macros-composants du système (GPU, FPGA, CPU). Des méthodes dites de refactoring peuvent être définies pour redéfinir les contraintes de placement sur les macro-composants et permettre une redistribution des calculs sur l'ensemble de la plateforme technologique considérée.

Au niveau architecture : la reconfiguration dynamique peut être intégrée dans le flot de conception pour faciliter la mise en oeuvre systématique de celle-ci. Des chaînes d'outils peuvent être aussi élaborées autour d'un modèle particulier comme les réseaux de neurones afin des proposer des optimisations plus adaptées.

Dans les sections suivantes, sont décrites plus particulièrement la méthodologie pour la gestion de reconfiguration dynamique, et la méthodologie pour la génération d'IP pour du deep learning.

2.4.2 Reconfiguration dynamique dans le flot de conception

La reconfiguration dynamique est généralement mise en place lorsque les ressources du FPGA ne sont pas suffisantes. Cela peut être le cas des applications Radar où le calcul intensif est accéléré sur FPGA. Concevoir des architectures reconfigurables de manière dynamique pose des contraintes pragmatiques lors de la mise en place d'applications. Les problèmes soulevés par la reconfiguration dynamique sont les suivants :

— garder une interface cohérente avec la zone reconfigurable en adaptant la présentation des données
— définir un critère de décision permettant de commuter d'une reconfiguration vers une autre. La définition de ce critère peut être de la responsabilité d'un expert du domaine. Le problème est alors de savoir comment exprimer ce critère et de l'intégrer au flot de conception.
— organiser un atelier de conception facilitant cette mise en place. La réflexion d'une approche méthodologique de conception intégrant les outils HLS existants peut reposer sur des concepts de méthodes de développement de type agile afin de faciliter les explorations des zones à reconfigurer.

Ces travaux autour de la reconfiguration partielle et dynamique dans le cadre des applications embarquées ont démarré avec les travaux de post-doc de Hanen Chenini avec le projet PICS [Che+15c][Che+15b]. La reconfiguration dynamique a été alors envisagée sur du traitement d'images et en utilisant un plateforme FPGA-SOC (Zedboard). Ces travaux précurseurs au projet HPeC ont été sources d'inspiration pour la mise en place des applications de traitement d'images dans le cadre de mission de drone autonome[Hir+18c]. L'étude de la configuration dynamique a été poursuivie à travers les projets autour du traitement Radar (projet LATERAL-RECONFIG) abordés lors de stage de master (stage Christophe Duhil en 2017) puis par la thèse de Julien Mazuet à partir de septembre 2018 financé par Thales.
Après avoir étudié des cas particuliers de reconfiguration dynamique pour les applications de poursuite Radar avec les filtres de Kalman et DFT/FFT [Maz+19][Maz+20], l'accent a été mis sur les aspects méthodologiques. L'intégration en amont de la capacité de reconfiguration est cruciale pour faciliter sa mise en oeuvre et notamment dans le contexte des grands groupe industriels comme Thalès. Une approche méthodologique reposant sur le concept des méthodes agiles a été proposée dans le contexte de la conception applications embarquées HW/SW alors que ce type de méthodes ne concerne que classiquement le contexte du logiciel. Cette approche permet de définir des interfaces entre différents types d'experts (expert application, expert architecture et expert logiciel) en séparant les domaines de compétence et en faisant progresser la définition du système par des contributions de chacun des experts via un processus itératif de raffinements successifs [Maz21].

2.4.3 Maitrise d'un flot pour architecture de type Deep Learning (ou apprentissage profond)

Dans la thèse de Tanguy Le Pennec, on s'est intéressé à la mise en place d'un classifieur d'images sous-marines. Le contexte de l'embarqué est défini par l'entreprise Thalès qui cherche à embarquer ce type de fonctions sur des petits sous-marins autonomes d'exploration de type AUV (Autonomous Underwater vehicle). La mise en place de ce classifieur a permis d'identifier quelques verrous scientifiques :

— les images sous-marines peuvent être dégradées par la faible luminosité, la turbidité et les courants des fonds marins. De plus les fonds sont en évolution continuelle, rendant la reconnaissance des lieux plus compliquée.
— les communications entre l'AUV et la plateforme d'amarrage sont très contraintes : les ondes acoustiques, qui assurent les communications, sont limités à la vitesse du son dans le milieu marin qui sont 200 000 fois plus lente que la vitesse de la lumière dans l'air. Comme elles offrent un débit limité, les communications restent alors peu nombreuses et limitées à l'essentiel.

— l'autonomie des véhicules est limitée et fortement liée à son énergie disponible pour accomplir sa mission. Il faut alors définir des systèmes efficaces en terme d'énergie et de calcul pour effectuer la mission de reconnaissance.

Pour répondre à ces verrous, des solutions à base de FPGA et de Deep learning adaptées au domaine sont actuellement à l'étude. Cette approche rencontre actuellement un vif intérêt dans la communauté scientifique comme souligné dans [Zha+22]. Cependant, les problèmes d'adéquation entre le modèle choisi (type réseaux de neurones convolutifs) et l'architecture cible restent des problématiques ouvertes dans la recherche de solutions temps-réel, peu consommatrices en énergie et adaptables aux ressources du support matériel cible.

Dans cette étude, une proposition de chaîne de conception sur FPGA et d'adaptation spécifique pour le traitement d'images sous-marin en UAV est en cours de développement. Cette chaîne repose sur les outils de HLS existants et utilise des formats standardisées de description de modèles (format Onnx) pour faciliter la migration de modèles sélectionnés à partir de framework comme PyTorch. Cette approche généralise l'approche récente proposée par 'HLS4ML' illustrée dans le figure 2.3. Les premiers résultats de ces travaux se trouvent dans [Le +20].

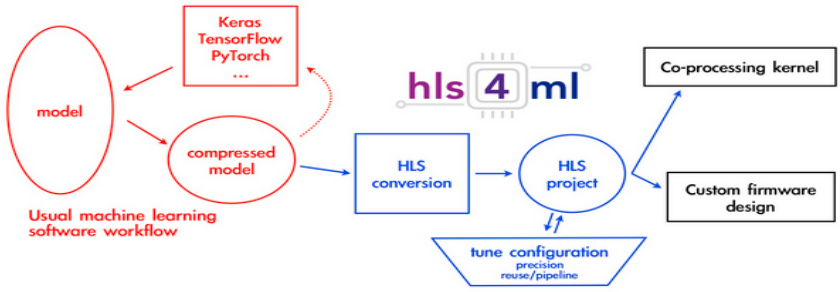

FIGURE 2.3 – Flot de conception HLS4ML intégrant des outils de HLS (source `https://fastmachinelearning.org/hls4ml/CONCEPTS.html| )`

2.5 Conclusion sur les aspects méthodologiques

Dans ce chapitre, on a présenté deux types d'approches principalement développées dans le but de concevoir des accélérateurs matériels : a) l'approche dédiée et b) l'approche intégrée. Le premier type d'approche a été mis en place alors que les outils de HLS commerciaux n'étaient pas disponibles sur le marché. Le deuxième type d'approche intègre les outils de HLS existants et proposent des extensions de ces outils. Dans la mise en place des outils, on peut retenir les points suivants :

— l'intérêt de l'approche objet : elle permet d'introduire un haut niveau de réutilisation des algorithmes au niveau du processus de conception, en se basant sur l'aspect

refactoring des algorithmes pour l'adaptation à divers types de données et de modèles.

— l'intérêt de l'approche intégrée : grâce à une la validation des extensions sur des outils commerciaux stabilisés, elle permet une meilleure acceptation des propositions dans le monde académique comme industriel.

Dans les aspects méthodologiques, on peut retenir les éléments suivants qui conditionnent le succès d'une acceptation par les concepteurs de circuits :

— il est préférable de proposer un point d'entrée standard ou usuel du domaine (ex spécifications en langage C, Python ou sous forme d'interface)
— il est souhaitable d'interagir avec l'utilisateur non expert afin d'éviter le phénomène de boite noire et permettre une explicabilité des résultats ou des transformations
— les transformations de code doivent rester le plus souvent possible correctes par construction
— les validations doivent être multi-niveaux afin d'assurer un traçage valide des transformations de code
— les solutions doivent être évaluées à chaque étape de raffinement et d'exploration

2.6 Perspectives sur la conception d'accélérateurs matériels

La conception d'accélérateurs reste un sujet d'actualité pour proposer des alternatives d'exécution efficaces en termes de performance et de consommation d'énergie.
Des perspectives possibles concernent

— la proposition d'ateliers permettant de définir aisément des IPs adaptées au contexte d'utilisation. La réutilisation de code est fortement souhaitable dans les étapes de transformations entre la description fonctionnelle et description matérielle afin d'éviter de regénérer une description matérielle déjà existante. Cette démarche de réutilisation de codes avec des retours d'expériences peut être exploitée à une plus grande échelle et peut généraliser la notion de réutilisation d'IP matérielle.
— l'élargissement du flot de conception en proposant des supports multiples (GPU, FPGA, CGRA, CPU, TPU) avec des passerelles pour faciliter les migrations d'un support vers un autre. La conception multi-supports permet de choisir le support matériel en fonction des familles de traitement que l'on peut identifier dans l'application principale.
— l'enrichissement des métriques classiques (consommation, latence, ressource) à différents niveaux en rajoutant des notions de fiabilité, sécurité, contexte environnemental, contexte durable...
— l'intégration de nouvelles technologies à base de Memristror[HK16] ou de spin-

tronique[KZZ19] pour faire du Computing-in-Memory (calcul dans la mémoire), afin d'éviter les acheminements des données qui peuvent constituer des goulots d'étranglement et des freins pour la course aux performances.

Chapitre 3

Diagnostic et synthèse de modèles probabilistes

Dans ce chapitre sont abordés les problèmes de sûreté de fonctionnement dans le cadre des véhicules autonomes. Des solutions logicielles originales sont mises en place dans le cadre du diagnostic pour élaborer des moniteurs intelligents. Pour embarquer ces éléments au sein de systèmes autonomes, des mises en oeuvres matérielles spécifiques pour plateforme à base de FPGA peuvent être proposées en vue d'un traitement temps réel des informations.

3.1 Contexte des travaux

A partir de 2012 et dans le cadre d'un nouveau partenariat entre l'ARCAA/QUT (Brisbane) et le Lab-STICC, j'ai participé au montage et à la mise en place du projet SWARMS (projet de recherche international financé par le CNRS) portant sur l'étude d'architectures adaptatives Hardware/Software pour des drones autonomes de type aérien. Pour ce projet, j'ai proposé d'utiliser des méthodes probabilistes pour répondre aux problèmes de sûreté de fonctionnement de ce type de véhicule. Ces méthodes seront reprises dans le projet ANR HPeC en 2015. Ces thématiques de recherche reposent essentiellement sur les modèles Bayésiens et ont été développées principalement dans la thèse de Sara Zermani et partiellement dans celle de Chabha Hireche. Les réseaux Bayésiens ont été aussi utilisés dans le projet RELIASIC (Projet CominLabs 2014-2019) pour caractériser les défaillances du GPS en fonction de son contexte environnemental.

3.2 Introduction et problématique

Les véhicules autonomes doivent faire face à des aléas de missions dus à des perturbations extérieures (température hors norme, vent non prévu, pluie...) et à des dysfonctionnements internes (capteurs défaillants, batterie dégradée,..). Cependant, ces aléas ne doivent pas mettre fin à la mission prévue s'il existe un moyen efficace de les gérer. Dans cette thématique, on s'intéresse aux modèles probabilistes qui permettent une prise en compte des aléas et une évaluation des défaillances pouvant concerner un capteur spécifique (GPS

Nature	Description
Limites physiques	Maximum poids en charge, vitesse, hauteur min/max
Limites temporelles	Temps de vol max, temps de réponse aux commandes ou temps d'acquisition à partir des capteurs, dégradation de la batterie au cours du temps, temps de vie de la batterie
Limites environnementales	Conditions climatiques (vitesse du vent, lumière ambiante, présence de poussière et de pluie, distance minimale par rapport à une zone d'habitants ou d'aéroports)
Limites comportementales	Actions effectuées par le pilote (si passage d'autonome à manuel)
Limites réseau	délai dû au réseau, gigue du signal, bande passante disponible, latence, disponibilité de connexion, congestion de trafic

TABLE 3.1 – Les limites identifiées des drones (source [All+19])

par exemple) ou tout autre élément du système embarqué au cours d'une mission. Dans ce chapitre, on considère aussi les versions embarquées des moniteurs pour satisfaire les contraintes temporelles et énergétiques définies dans le cadre de mission d'un drone autonome. Des mises en oeuvre matérielles pour FPGA sont proposées pour accélérer le diagnostic et pour réaliser des versions basse consommation énergétique. Ces modules peuvent être vus ici comme des IPs (Intellectual Property) produites en utilisant des outils HLS commerciaux, la plus value de cette approche se situant dans le front-end (pré-traitement) proposé et l'interface vers un outil HLS. Ce chapitre n'aborde pas l'intégration du diagnostic au niveau de la prise de décision qui est présenté dans le chapitre 4.

3.3 Résilience et autonomie : cas des UAV

Les véhicules autonomes doivent prévoir un certain nombre d'alternatives pour satisfaire leur mission. Ces alternatives permettent de faire face à un certain nombre d'aléas en fonction des limitations/contraintes dues aux véhicules. Nous nous sommes intéressés plus particulièrement au cas des drones aériens pour lesquels les limitations d'ordre physique, temporel, environnemental, comportemental ou structurel existent. Ces limites sont décrites dans la table 3.1. Les aléas auxquels les UAVs peuvent être soumis peuvent être d'origine interne ou externe. Ces aléas font référence à des rapports d'incidents ou d'accidents issus de base de données identifiées et citées dans [All+19].

Pour détecter les cas de figure de dysfonctionnements liés aux différents aléas, il est nécessaire de mettre en place des modules de diagnostic qui puissent être au coeur de la boucle de fonctionnement d'un drône, soit dans la boucle OODA (Observation-Orientation-Decision-Action)[Fad95]. L'observation est effectuée par le capteur qui renvoie des données issues de cette observation au système. Ces données sont ensuite filtrées, analysées pour orienter la décision qui va définir l'action à exécuter. Les modules de diagnostic se trouvent entre l'observation et le traitement pour mieux orienter les décisions, parfois au plus proche

Source	**Type**	**Exemple**
Externe	Interférence	Interférences électromagnetiques, humaine (écoute de signal radio), interférences de communication
Externe	Conditions env.	vent, température, atténuation atmosphérique, glace, précipitation, luminosité
Externe	Obstacles	obstacles fixes (arbre, cables électriques, buildings) et obstacles dynamiques (oiseaux, voitures)
Externe	Env. navig.	Perte/erreur signal GPS, GPS Spoofing, erreur système de navigation, erreur d'altitude, waypoint erronné
Externe	Env. traffic	drone ou avion à proximité, zone de traffic à accès limité
Externe	Env. elec.	zone RF avec un haut traffic radio, phénomène electrostatique
Externe	Communication	congestion du réseau, délais ou indisponibilité du réseau, gigue
Externe	Facteur humain	manquement aux consignes de sécurité, attaque de sécurité, erreur de pilotage
Interne	Facteur Mécanique	défaillance mécanisme de fermeture, défaillance de moteur
Interne	Thermique	givrage, explosion
Interne	Electronique	perte de puissance, defaillance de propulsion, saturation
Interne	Algorithmique	erreur de vérification, de décision, réponses avec délai, boucle infinie
Interne	Facteur technique	baisse de batterie, cellule de batterie défaillante, malformation technique, cycle de charge inapproprié, perte de controle, perte de transmission
Interne	Logiciel	défaillance du système de controle, erreur de l'autopilote, bugs dans le code, erreur dans le process, system de visio en panne
Interne	Matériel	erreur sur CPU, matériels avioniques, capteurs

TABLE 3.2 – Sources des aléas pour un drone (source [All+19])

des capteurs en étant intégrés dans les pré-traitement des données.

3.3.1 Modèles et approches pour le diagnostic

Le diagnostic rentre classiquement dans les approches dites FDIR (Fault Detection Isolation and Recovery) et peut reposer sur différentes approches[YWL12][GCD15] : basées sur des modèles (model-based), ou sur du traitement de signal (signal-based), sur l'analyse de données (data-based or knowledge-based).

— L'approche signal-based utilise des signaux pour diagnostiquer de possibles fautes ou anomalies en comparant le signal détecté avec l'information issue d'un système non erroné. Les fautes sont examinées à travers le signal mesuré et le diagnostic est basé sur l'analyse des symptomes/patterns. Cependant, il est difficile de construire un modèle mathématique précis ou d'obtenir des patterns de signaux significatifs. Les caractéristiques des signaux peuvent être définies dans le domaine temporel ou/et fréquentiel. Les méthodes peuvent utiliser des caractéristiques extraites des images (SIFT), l'analyse de signatures dans le domaine fréquentiel (MCSA), ou des décompositions en ondelettes (WT) [GCD15].

— Dans l'approche model-based, l'attention est donnée à l'établissement d'un modèle mathématique. Les modèles reposent sur la caractérisation mathématique du système sous observation pour analyser les défaillances et dégradations. Cette approche repose sur la génération de résidus entre les valeurs du capteur mesurées et les valeurs de sorties prédites par le modèle mathématique. On trouve dans cette catégorie les filtres de Kalman (KF) ou des variantes comme l'extended Kalman (EKF), filtre à particule (FP) et les méthodes Bayésiennes (MB).

— L'approche data-driven a besoin d'un large ensemble de données historiques plus que des modèles ou des patrons de signaux. L'ensemble des données disponibles permet de caractériser le modèle sous-jacent par des mécanismes d'apprentissage supervisé ou non supervisé. On trouve ici les modèles comme les arbres de décision(DT), les arbres de défaillances (FT), les systèmes experts(ES), les systèmes à base de logique floue (FL), à base de réseaux de neurones (ANN, DNN), de réseaux de Petri, les chaînes de Markov (MC) et aussi les réseaux Bayésiens (BN). Dans cette approche, on trouve aussi les regressions linéaire ou non-linéaires, les réductions de dimension (PCA), les algorithmes de clustering (ex k-Means). Cette approche data-driven peut être statistique ou non-statistique [ZCD15] ou une combinaison des deux.

Les avantages et inconvénients de quelques approches ont été analysées dans [YWL12][Ran+22] montrant l'intérêt pour l'approche data-based intégrant des méthodes dites intelligentes capables de mieux capturer la complexité du système à diagnostiquer. Le tableau 3.3 donne un aperçu rapide des différentes approches et leurs intérêts respectifs.

Les modules de diagnostic s'intègrent dans les mécanismes de management du système

Type d'approche	Méthodes utilisées	Avantages	Inconvénients
Signal-based	analyse temporelle (SIFT) analyse fréquentielle (MCSA) analyse mixte(WT)	Mise en place plutôt facile	Possibilité de mauvais reports et fausse alarme
Model-Based	Filtre de Kalman (KF) Extended KF (EKF), Filtre à particules (FP), MB	Bonne précision, Extrapolation possible	Bonne connaissance du domaine, complexité modèle
Data-based	Regression, Clustering, reduct. dim. (PCA), decision Tree (DT), Fault Tree (FT) chaîne de Markov (MC), BN, logique floue (FL), système à base de règles, réseaux de neurones (ANN)	Peu de connaissance nécessaire du domaine, Algorithme facile à développer	Gros volume de données physiques nécessaire, données sur les défaillances difficiles parfois à obtenir

TABLE 3.3 – Différentes approches pour le diagnostic de fautes version revisée de [Ran+22]

de santé des véhicules autonomes Integrated System Health Management (ISHM) qui utilisent principalement des approches model-based et/ou data-based [FW18] pour une gestion intelligente du système de santé. Le principal défi des ISHM intelligents réside dans leur capacité à fournir en temps-réel des conseils de planification et à effectuer des ajustements face à des composants défaillants avec des reconfigurations du système ou des modifications du profil de la mission. Ceci constitue encore en 2022, un des principaux objectifs pour assurer la sécurité des systèmes autonomes [Ran+22].

3.3.2 Réseaux Bayésiens pour la sûreté de fonctionnement des véhicules autonomes

Dans les travaux que j'ai menés sur la sûreté de fonctionnement, j'ai choisi d'utiliser principalement le modèle des réseaux Bayésiens (RB ou BN pour Bayesian Network). Ce modèle a été exploité notamment dans les travaux de thèse de Sara Zermani et Chabha Hireche, commencés respectivement en 2013 et 2015. Un réseau Bayésien est un modèle graphique probabiliste permettant de gérer de l'information incertaine en proposant des mécanismes d'inférence pour évaluer les probabilités du diagnostic. Les principales motivations concernant l'utilisation de ce modèle sont les suivantes :

— **modèle probabiliste** : choix d'un modèle probabiliste pour faciliter l'intégration des incertitudes liées aux aléas internes ou externes

— **popularité du modèle** : modèle déjà largement utilisé pour du diagnostic dans différents domaines comme le domaine médical, robotique, du processus industriel, financier. On trouve dans [Cai+18] en 2019, une confirmation de l'intérêt de l'usage des BN pour l'évaluation de la fiabilité pour différents type d'objets correspondant à du matériel, à des structures, à du logiciel ou à de l'humain. La popularité de ce

Analyse des risques	aide dans la prédiction du coût (conséquences) des actions ou du dysfonctionnement
Détection de fautes et diagnostic	aide dans l'identification des composants dysfonctionnant et les raisons [SMM11]
Prédiction	aide à l'anticiper des actions face aux évènements changeant dans l'environnement (associé à de l'incertitude)
Planification de mission et sélection de but	aide à la modélisation et analyse de situations complexes
Interaction humain-véhicule	aide à la modélisation de leur interactions
Conscience de la situation	aide à la fusion d'information entre différentes sources pour détecter des comportements anormaux
Exigence de sécurité	aide pour intégrer des standard de sécurité issus d'organisations gouvernementales
Prise de décision	aide à l'évaluation de l'exécution d'actions sélectionnées, en prenant en compte le niveau d'autonomie, la confiance, la responsabilité, les causes et les conséquences

TABLE 3.4 – Liste des cas de figure pouvant être modélisés par des réseaux Bayésiens pour les IAVs (source [DMC19])

modèle a permis aussi d'attirer l'attention sur les méthodes d'inférence et toutes les méthodes permettant de paramétrer le modèle. Dans le cas des véhicules autonomes intelligents IAV (intelligent autonomous vehicle), les réseaux Bayésiens peuvent être utilisés à différents niveaux comme le montre l'étude récente publiée par Torres en 2019[DMC19]. Les exemples d'utilisation sont repris dans le tableau 3.4.

— **efficacité du modèle** : la supériorité des méthodes BN par rapport aux méthodes conventionnelles d'analyses de risque telles que la méthode Bowtie ou arbre de défaillance a été prouvées dans la littérature [KKA12 ; KKA11 ; Bob+01]. Les réseaux Bayésiens permettent de définir de manière plus ou moins detaillée des corrélations entre les éléments du système et des facteurs extérieurs au système comme l'environnement, permettant de donner une vision globale et holistique du problème. Les aspects temporels peuvent aussi être pris en compte.

Les réseaux Bayésiens nous ont permis de répondre au moins partiellement aux défis/verrous suivants :

— Définir un modèle puissant mais facile à mettre en place pour des non-experts, sans avoir besoin d'une connaissance à priori du modèles BN
— Définir des méthodes d'inférences temps-réel, non intrusives pour faire du diagnostic
— Proposer des techniques de recouvrement en ligne pour la correction d'erreur
— Proposer des méthodes d'auto-adaptation pour effectuer à la demande des mises à jour du modèle

— Proposer des méthodes de classification de défaillance, d'attaques de sécurité pour contribuer à la définition d'un système de santé plus fiable au sein du drones
— Proposer des méthodes qui puissent s'adapter au cas d'une organisation de mission en équipe ou en essaim de drones.

Avant d'illustrer quelques uns de ces défis dans les sections suivantes, une présentation succincte des réseaux Bayésiens est faite dans la section qui suit.

3.4 Petit focus sur les réseaux Bayésiens

3.4.1 Les réseaux Bayésiens : comment ça marche et pourquoi ?

Les BN sont des modèles graphiques probabilistes utilisés pour comprendre et contrôler le comportement d'un système [Cow06]. Les noeuds dans un tel réseau représentent des variables aléatoires, les probabilités conditionnelles de ces variables sont données par les tables de probabilités (CPTs) associées au noeud. Les arcs du réseau indiquent les dépendances conditionnelles. La structure du réseau Bayésien est définie par la structure de graphe reliant les différents noeuds via les arcs identifiés. Les paramètres du BN sont associés aux tables CPT des noeuds. Un exemple simple de BN est donné dans la Figure 3.1. Le noeud UAV Altitude (U_A) représente le status de l'altitude décrivant si l'altitude de l'UAV est croissante ou décroissante. Les noeuds Barometre (S_B) et Altimètre (S_A) représentent les capteurs qui observent l'UAV prenant de l'altitude ou réduisant son altitude. Le noeud U_A influence les lectures du capteur, d'où les arcs liant le noeud U_A aux noeuds S_A et S_B. Les tables CPT pour le capteur (S_A), par exemple, devraient être lues comme suit : si le status U_A est montant, alors la probabilité lue augmente à la valeur 0.7.

Le diagnostic est obtenu en utilisant le mécanisme d'inférence basé sur l'observation de S_A ou de S_B, et en calculant la probabilité a posteriori pour le noeud U_A se trouvant dans un statut d'altitude croissante. Cette probabilité se calcule en utilisant le théorème de Bayes donné par l'équation (3.1).

$$P(A|B)P(B) = P(A \cap B) = P(B|A)P(A) \tag{3.1}$$

On trouve (Equation (3.2)) :

$$P(U\_A = Inc|S\_A = Inc) = \frac{P(S\_A = Inc|U\_A = Inc)P(U\_A = Inc)}{P(S\_A = Inc)} \tag{3.2}$$

$$P(U\_A = Inc|S\_A = Inc) = \frac{0.7*0.5}{0.7*0.5+0.2*0.5} = 0.778$$

Avec une observation sur l'altimètre S_A, on peut dire que la probabilité du status d'altitude d'être dans l'état croissant est égal à 0.778 comme montré dans la Figure 3.1b.

Si par exemple, on ajoute des observations du baromètre donnant une valeur croissante, la probabilité $P(U\_A = Inc|S\_A = Inc, S\_B = Inc)$ augmente jusqu'à la valeur 0.969. Après cet exemple illustrant le mécanisme d'inférence, nous abordons dans le paragraphe suivant la mise en place du réseau Bayésien.

3.4.2 Mise en place du réseau Bayésien

Pour définir le réseau Bayésien dans le cas d'identification de faute, plusieurs points sont à aborder : la définition de la structure du réseau, la détermination des paramètres (CPT), l'algorithme d'inférence pour le calcul de probabilité permettant d'identifier de la faute, la validation du réseau basée sur des scénarios.

Structure du réseau : l'identification de structure est établie à partir des relations cause-conflit entre les divers éléments. Cette structure peut être établie par des experts du domaine ou en utilisant des algorithmes d'apprentissage de structure.

Paramètres du réseau : Les tables conditionnelles du réseau et les tables à priori peuvent être définies par des experts ou des algorithmes d'apprentissage de paramètres.

Algorithme d'inférence : on distingue deux classes d'algorithmes pour l'inférence ; algorithmes à solutions exactes ou à solutions approchées. Ces algorithmes d'inférence sont utilisés pour le calcul de la probabilité a posteriori permettant d'identifier la faute.

Validation et vérification : Déterminer des cas de validation permettant de mettre en avant l'apparition de fautes et simulation des différents cas de figures avec la mise en place de scénarios.

Ces différents points sont décrits dans la figure 3.2. avec des possibilités de révision à chacune des étapes en fonction des validations choisies. On peut aussi trouver les détails concernant l'apprentissage et l'inférence des réseaux Bayésiens dans [Pea88] [Naï+07] [Mur02][Dar09] [Ler06] [DSA11] [Cai+18].

3.4.3 Estimation, prédiction et mise en place de solution de compensation

Pendant les missions UAV, l'évaluation de l'état de santé des composants du système (capteurs, actionneurs, etc.) est effectuée à partir des données du capteur [Zer+17b]. D'autre part, l'état de qualité de service (QoS) des applications exécutées à bord, telles que l'application de suivi de cible, peut également être surveillé pour assurer le succès de la mission. La Figure 3.3 donne un extrait de la modélisation de la surveillance de l'application tracking, une version plus complète se trouve dans la Figure 3.7 ou dans [Hir19]. Pour cette application, nous énumérons les différents problèmes potentiels

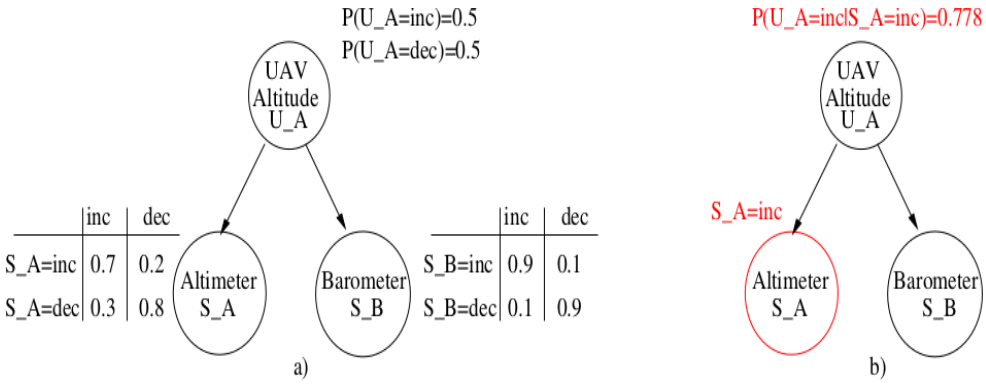

FIGURE 3.1 – **(a)** Définition du BN avec ses paramètres (CPTs) ; **(b)** Illustration du mécanisme d'inférence

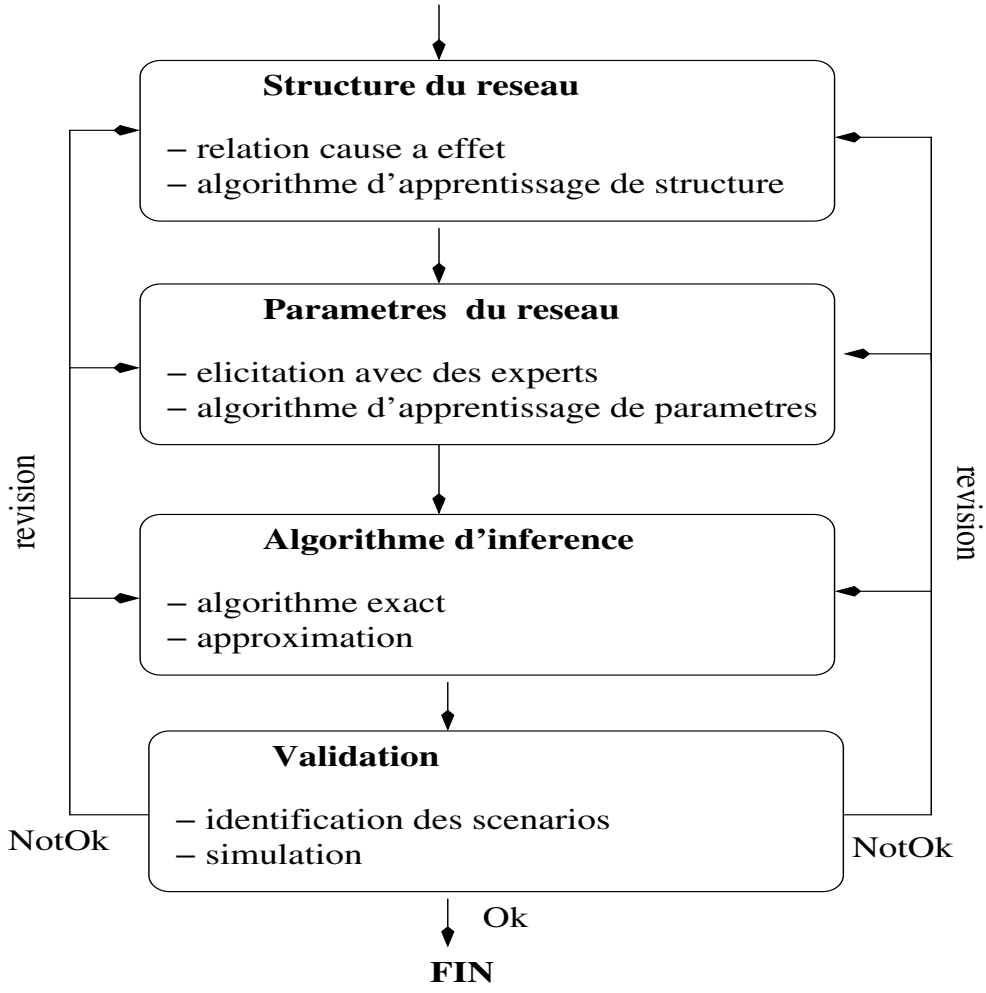

FIGURE 3.2 – Flot pour la mise en place d'un réseau Bayésien-adaptation de [Cai+18]

qui peuvent survenir. Dans cet exemple, l'un des problèmes potentiels est la vibration qui peut affecter la qualité du tracking. La vibration peut se produire lorsque l'IMU montre une déviation ou lorsqu'un vent fort est détecté. Le BN permet d'évaluer la QoS de l'application en fonction des preuves de la déviation de l'IMU ou de la détection du vent comme décrit dans Figure 3.4.

Le BN peut également aider à atténuer l'erreur en proposant une solution de compensation qui serait la plus adéquate lorsqu'une erreur est détectée. Dans ce cas, le BN inclut la proposition d'atténuation potentielle. L'identification d'une solution compensatoire la plus adéquate peut être mise en place en évaluant par inférence les probabilités des adaptations proposées en fonction du contexte identifié par les capteurs. La solution compensatoire ayant la plus haute probabilité d'activation est alors choisie pour assurer un certain niveau de qualité de service de l'application. Sur l'exemple de la Figure 3.5, on peut voir qu'il est nécessaire, dans le cas d'une vibration et de vent fort, d'activer la stabilisation pour obtenir une QoS élevée : en effet, la probabilité d'activation de cette fonction est évaluée à "haute" (0.97) dès lors que la QoS est elle-même supposée haute. Le choix de l'activation dépend du contexte, en effet la stabilisation doit être activée dans un contexte de vent et de vibration détectée au niveau de l'IMU. Dans un contexte différent notamment sans vent et sans vibration au niveau de l'IMU, l'application de stabilisation n'est pas nécessaire comme montré dans Figure 3.5.

Les réseaux Bayésiens peuvent encapsuler d'autres informations et contraintes concernant le fonctionnement ou l'évaluation de la santé des applications ou des éléments matériels (capteurs ou calculateurs) qui peuvent être embarqués. Ces modèles permettent donc

- de définir des estimateurs de l'état de santé des capteurs en évaluant la probabilité de bon fonctionnement du système
- de faire de la prévention d'erreurs en évaluant des alternatives possibles
- d'intégrer des contraintes diverses (ressources, performance et énergie) pour guider le choix de la solution embarquée

Les évaluations de ces modèles reposent sur des observations de l'environnement et des contraintes imposées par le système embarqué qui sont intégrées comme des observations.

Dans le contexte des drones autonomes, les réseaux Bayésiens peuvent être utilisés pour

- Identifier des problèmes au niveau des capteurs (ex GPS), des éléments constitutifs du drone (ex Batterie)
- Identifier des variations d'applications plus appropriées au contexte de la mission. Par exemple dans l'application de 'suivi de cible' , on peut élaborer un BN pour prendre en compte le contexte environnemental (vent fort, vitesse du drone, vitesse de la cible, luminosité et taille de l'image), les informations de suivi associées aux retours de capteurs et retours des traitements applicatifs afin de mieux réguler la qualité de service de cette application.

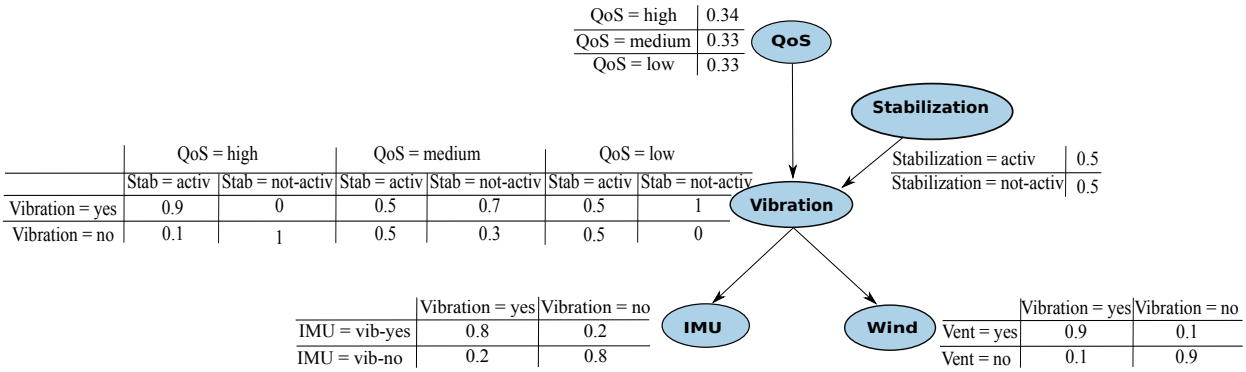

FIGURE 3.3 – BN pour évaluer la QoS en fonction des vibrations.

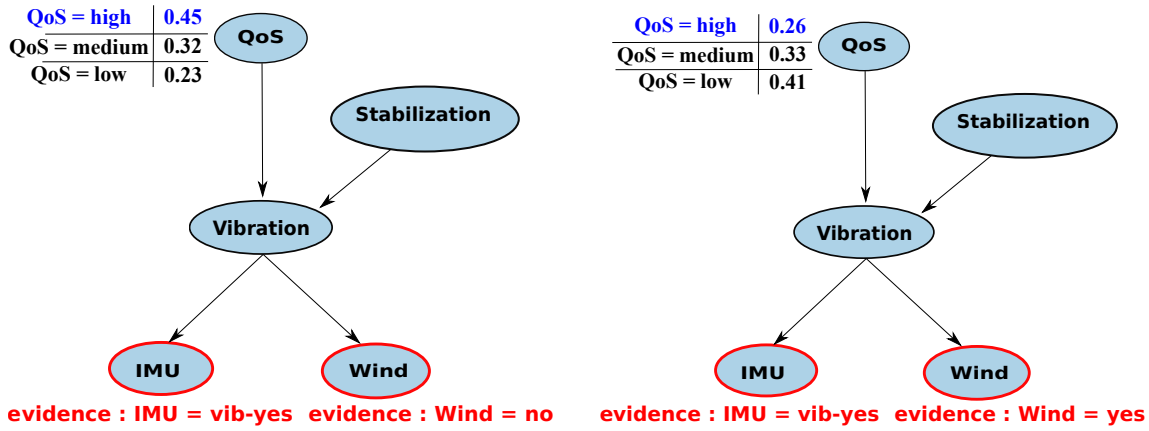

FIGURE 3.4 – Evaluation de la probabilté de la QoS en fonction du contexte observé.

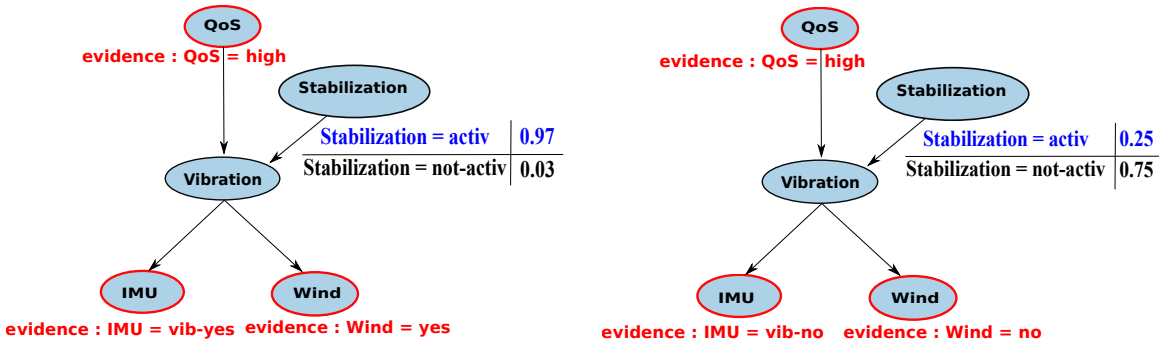

FIGURE 3.5 – Probabilité d'activer la stabilisation afin de maintenir une bonne QoS de l'application en fonction des observations sur l'IMU et au niveau du vent

La mise en place de ces BN peut s'avérer fastidieuse. Pour faciliter cette mise en place, une approche de conception système s'appuyant sur des patrons de conception a été envisagée.

3.4.4 Approche de conception dirigée par les patrons de conception

Les structures spécifiques des BN capables de spécifier les moniteurs intégrant éventuellement la compensation d'erreur peuvent être définis automatiquement à partir d'une table de type AMDEC (analyse des modes de défaillance, de leurs effets et de leurs criticités) (ou FMEA en anglais pour Failure Modes and Effects Analysis) en utilisant une approche pilotée par des patrons de conception. La version intégrée de ces moniteurs BN nécessite d'élaborer de nouveaux outils de conception pour les implémenter sur FPGA. Nous avons proposé un flux de conception complet des tables FMEA vers l'implémentation FPGA dédié à l'aide à l'élaboration de moniteurs BN. Nous avons défini différents patrons qui peuvent intégrer des contraintes d'environnement, des contraintes système, des contraintes applicatives basées sur la qualité de service et les alternatives possibles. Les patrons identifiés sont associés à des structures spécifiques du BN qui peuvent être élaborées automatiquement à partir de tableaux de type AMDEC. Les trois modèles principaux que l'on a identifés sont : le $FMEA\_HM$, le $FMEA\_MIT$ et the $FMEA\_EMB$, représentant respectivement le diagnostic contextuel, la proposition d'atténuation des erreurs intégrant éventuellement les contraintes liées au contexte embarqué (par ex : des contraintes de performance ou de ressources matérielles).

Un exemple de patron est donné dans la figure 3.6. Ce patron est élaboré à partir de la table $FMEA$ donnée dans la table 3.5 et représente un patron de type $FMEA\_HM$. Cette table peut être augmentée avec les propositions de compensation d'erreur dans le cas du patron $FMEA\_MIT$ comme proposé dans l'exemple de la table 3.6. Le BN correspondant à cette spécification est celui de la figure 3.7.

3.4.5 Raffinement du modèle

L'approche de conception utilisant les patrons à partir des tables $FMEA$, permet d'aider à la construction des BN mais cette approche s'intéresse à la structure du réseau et non à ses paramètres.

Généralement, ces probabilités sont définies par un expert du système. Cependant, cette expertise peut être incomplète ou impossible à réaliser dans certains domaines ou systèmes. Dans ce cas, il est nécessaire d'utiliser une méthode d'apprentissage pour les paramètres BN [DSA11]. Il existe différentes méthodes d'apprentissage pour les paramètres BN dans le contexte de données complètes [GD04] ou de données incomplètes [Fri98]. La mise en place des moniteurs, comme dans le cas de l'état de santé du système autonome, peut utiliser les bases de données mais qui sont souvent incomplètes. Les données manquantes peuvent être

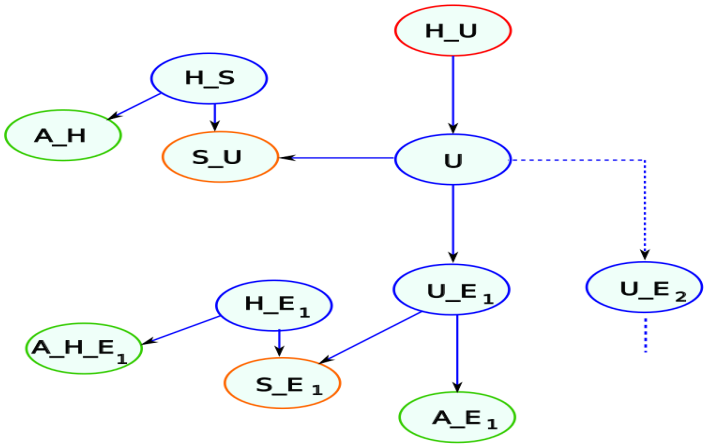

FIGURE 3.6 – Patron de type $FMEA\_HM$ où les noeuds S_U, S_Ei représentent les capteurs (observateur matériel ou logiciel), les noeuds U, U_Ei sont les états internes possiblement affectés par une erreur Ei, le noeud H_U représente l'état de santé du système, les noeuds H_S et H_Ei représentent l'état de santé du capteur et A_H, A_H_Ei représentent les contextes environnementaux

TABLE 3.5 – Table FMEA avec une erreur de type Health Management (pattern $FMEA\_HM$).

Type d'erreur	**Monitoring**	**Contexte env. (pour type erreur)**	**Contexte env. (pour Monitoring)**
U_E_i	S_E_i	A_E_i	$A\_H\_E_i$

TABLE 3.6 – Exemple de table $FMEA$ pouvant être transformé par une approche dirigée par les patrons en BN

Erreurs	**Possible Observations**	**Contexte Environemental**	**Solution (Algorithme)**
Vibration	IMU (Inertial Measurement Unit) Vibration capteur	Vent Vibration	Activation de la stabilisation
Suivi de cible perdu	Modèle basé sur : nombre de caractéristiques détectée (Harris) [KMM12]	Vitesse du drone variations de la luminosité	Amelioration du contraste
Vecteur de mouvement perdu	Modèle basé sur : vecteur de mouvement entre 2 images [KMM12]	Vitesse cible R.O.I petite (Région d'intérêt)	Augmentation de la taille R.O.I

dues à des défaillances de capteurs, des mesures inexactes des données des capteurs, etc. Dans ce cas, il est possible d'apprendre les paramètres BN à partir des bases de données en utilisant l'algorithme EM (Expectation Maximisation) [Tem+16]. Comme cet algorithme peut considérer la structure du BN, la phase d'apprentissage peut être simplifiée. Dans [HDD17], nous avons montré que nous pouvons réduire le temps d'apprentissage des paramètres si la structure du BN est connue à l'avance en se limitant à un ensemble de paramètres significatifs localisés sur les noeuds feuilles du BN.

3.5 Concevoir des moniteurs intelligents embarqués

La conception de moniteurs pour l'embarqué doit relever les défis suivants :

— Mise en place hors-ligne de moniteurs en proposant une chaîne de conception pour ces moniteurs qui permet une implémentation sur support reconfigurable. Le choix du support reconfigurable permet de favoriser une implémentation parallèle.
Des méthodes de compilation efficaces sont alors nécessaires pour faciliter la mise en place d'une architecture parallèle.

— Elaboration d'une stratégie de validation. Il est nécessaire de vérifier que l'intégration au niveau du drone soit valide et qu'elle puisse effectuer correctement les détections. Cette validation peut se faire par des expérimentations sur un drone en charge de mission ou dans un environnement de simulation complètement virtuel ou partiellement virtuel. Dans le cas de simulation partiellement virtuel, on intègre quelques éléments hardware dans l'environnement de simulation en utilisant la technique HIL (Hardware in the loop).

3.5.1 Atelier de conception dédié et intégré

La mise en place d'une version embarquée de réseaux Bayésiens pour le diagnostic ou l'aide à la recherche d'application adaptée au contexte, repose sur la recherche d'une version parallèle du calcul d'inférence à effectuer. L'atelier de conception proposé qui assure cette version embarquée possède les caractéristiques suivantes :

— il ne demande pas de connaissance a priori concernant le modèle sous-jacent afin de faciliter leur utilisation. Il faut alors que leur spécifications soient simples.

— il peut facilement s'interfacer avec des outils de HLS pour une mise en oeuvre rapide sur FPGA ou FPGA-SoC ou des compilateurs classiques pour une exécution sur processeur standard.

Plusieurs points sont à prendre en compte pour l'interfaçage :

1. En amont, il faut définir une forme interne à partir de la spécification de haut niveau pour laquelle il sera plus facile d'exprimer du parallélisme, de proposer des découpages fonctionnels, d'explorer le typage des données.

2. A l'interface, le paramétrage de la forme intermédiaire doit être compatible avec les outils de synthèse HLS existants et intégrer des directives de synthèse de haut niveau pour gérer efficacement la mise en oeuvre matérielle prenant en compte les contraintes temporelle et de ressources du support cible.

Plusieurs algorithmes de compilation peuvent être utilisés pour effectuer l'inférence exacte. On peut citer parmi ces algorithmes, les méthodes à base d'arbre de jonction(JT) et les méthodes dites 'Circuits Arithmétiques'(AC)[Dar03]. Les compilations JT sont celles classiquement utilisées dans les differents frameworks proposés pour faire du calcul d'inférence sur les BNs (par ex Genie, BayesiaLab, BNToolBox de Matlab). Néanmoins, les formulations de type AC permettent une formulation sous forme d'arbres d'opérateurs arithmétiques qui est plus facilement parallélisable. La définition d'un outil de conception automatique pour les moniteurs Bayésiens repose sur deux points clés :

— le premier est la capacité de fournir une interface avec des outils de synthèse de haut niveau afin de réduire les efforts d'ingénierie pour sa mise en place sur support reconfigurable.

— le deuxième concerne les optimisations, qui peuvent être de deux types : indépendante ou dépendante de la mise en oeuvre matérielle. La première catégorie d'optimisation est spécifique aux calculs dans les structures AC. Elle est associée au typage des données et au découpage hiérarchique des calculs. La deuxième catégorie d'optimisation définit les directives de HLS, d'interface et de mémoire les plus appropriées.

La méthodologie de conception pour la partie hors ligne que l'on a proposée est illustrée dans la Figure 3.8.

3.5.2 Du concept à la réalisation

La mise en place des moniteurs Bayésiens a pu être envisagée dans deux cadres de projet : dans le cadre du projet SWARMS (PICS-Projet international de coopération scientifique avec la QUT - Queensland Technological University) pour la période 2013-2015, puis dans le cadre ANR HPeC pour la période 2015-2019. Deux types de moniteurs ont pu ainsi être mis en place : des moniteurs associés aux éléments physiques incluant les capteurs (GPS, Batterie) et des moniteurs associés aux applications (application de suivi de objet par exemple).

Les techniques de validation assurant une intégration cohérente sur le système du drone sont effectuées à plusieurs niveaux et de manière incrémentale. La démarche de validation généralement adoptée est la suivante :

1. Mise en place de scénarios permettant de valider les différents moniteurs (ex avec une modélisation Matlab)

2. Mise en place d'une infrastructure pouvant être exécutée sur le système du drone

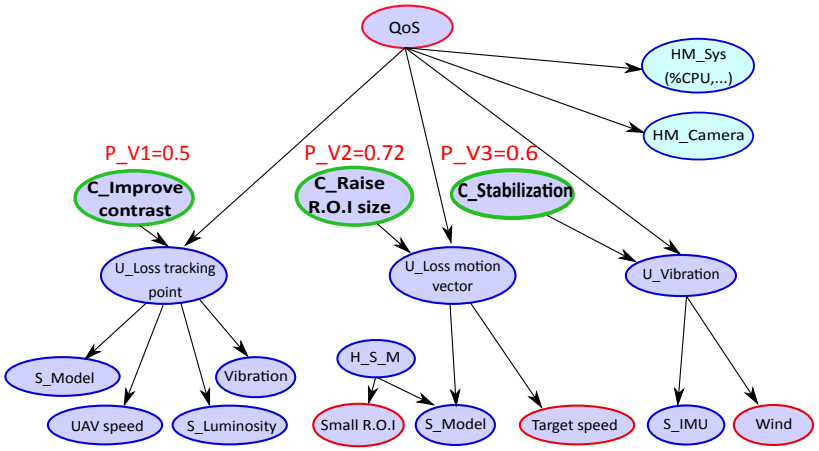

FIGURE 3.7 – BN associé à l'application de suivi de cible avec solutions de compensation d'erreur afin de maintenir une bonne qualité de service

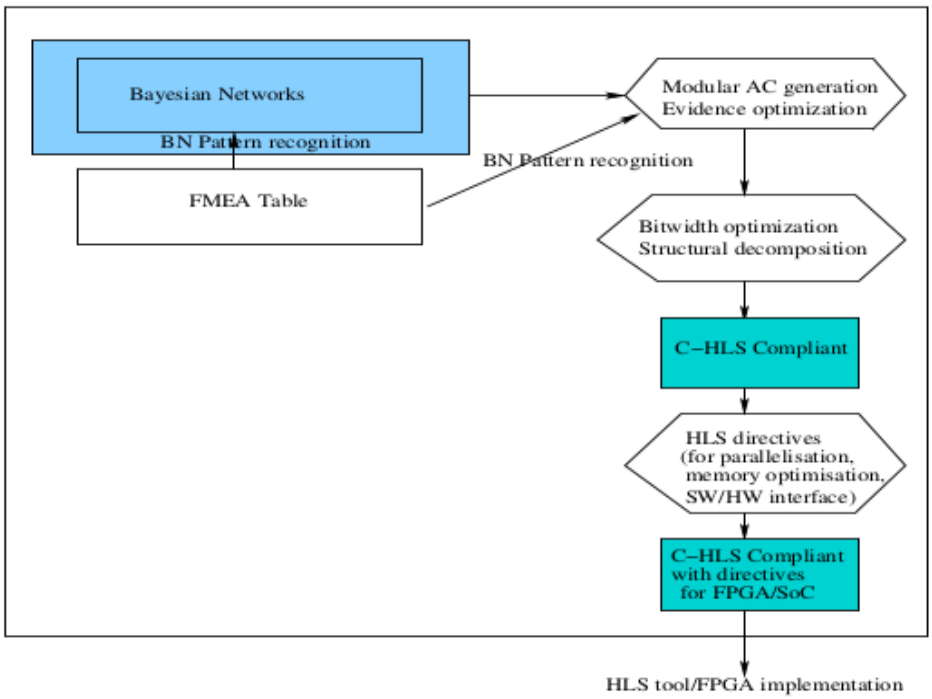

FIGURE 3.8 – Méthodologie de conception pour la mise en oeuvre sur FPGA des moniteurs BN à partir de table FMEA (F2F BNTool).

organisé en utilisant ROS (Robot Operating System). ROS est constitué d'une collection d'outils et de librairies classiquement utilisée pour modéliser les communications et les interactions entre les différentes applications (tâches) et entre les applications et les capteurs embarqués sur le robot.

3. Intégration d'éléments hardware dans l'infractucture ROS et cohabitation avec un environement de simulation (en 2D ou 3D) du contexte de la mission reposant sur des simulateurs connus du domaine de la robotique comme Gazebo ou SITL.

Cette démarche a pu être validée dans le projet HPeC dans le cadre de la validation du mission manager complet qui utilisait des moniteurs Bayésiens applicatifs, elle est décrite de manière plus détaillée dans la thèse [Hir19].

3.6 Conclusion/bilan de l'approche par réseaux Bayésiens

Dans ce chapitre, nous avons abordé la thématique de la sûreté de fonctionnement dans le contexte des drones autonomes. Plusieurs aspects ont été explorés :

— aspect modèle avec le choix des réseaux Bayésiens pour évaluer les défaillances d'un système et proposer des solutions compensatrices. Ce modèle a retenu mon attention car il permet une mise en lumière des corrélations entre les différents éléments du système en y associant leur contexte d'utilisation (caractérisation matérielle, logicielle du système du drone) et le contexte environnemental externe de la mission (facteurs climatologiques, situation géographique). Le choix de ce modèle a été conforté par des références solides dans le domaine (ex celui de la NASA [Sch+15]) et surtout confirmé par les validations de nos expérimentations.

— aspect atelier de conception avec une aide à la mise en place des modèles pour les non-experts des réseaux Bayésiens. Une réflexion sur les points d'entrée de cet atelier a été menée pour qu'un ingénieur ou chercheur familier avec la sûreté de fonctionnement puisse facilement générer les moniteurs Bayésiens souhaités. Si l'atelier proposé facilite la mise en place de la structure BN de ces moniteurs, la mise en place des paramètres n'est pas aussi simple et nécessite de passer par des phases d'apprentissage à partir d'ensembles de données. Pour certains types de structure, on a montré que cet apprentissage peut se limiter à certains noeuds sans modifier pour autant la précision du réseau[HDD17].

— aspect intégration dans le système embarqué et validation. La chaine de conception propose une méthodologie permettant de générer des accélérateurs matériels sur FPGA pour répondre aux injonctions de performance des moniteurs élaborés. La validation est essentiellement effectuée par simulation en s'appuyant sur des scénarios de mission.

3.7 Perspectives sur le diagnostic

L'aspect sûreté de fonctionnement est un élément incontournable dans la définition des systèmes autonomes. Il devient indispensable de définir des moniteurs intelligents capables d'analyse une situation en fonction du contexte, et de poser un diagnostic sur celle-ci : normale, anormale en premier lieu pour identifier par la suite le cas d'anomalie.

Les modèles Bayésiens offrent un modèle puissant permettant de définir des corrélations entre les éléments qui permettent d'envisager des modèles efficaces. Evidemment, ce ne sont pas les seuls modèles possibles pour détecter des anomalies mais ils présentent le gros avantage de donner des solutions explicables et interprétables.

Dans les travaux présentés ici, les anomalies relèvent principalement du domaine de la sûreté de fonctionnement mais elles pourraient correspondre à des malveillances. Les perspectives correspondant à cette thématique sur le diagnostic sont :

— l'introduction de l'aspect sécurité. Les problèmes de sécurité peuvent être identifiés grâce à des IDS (systèmes d'identification d'intrusion). Ces IDS peuvent utilisées des classifieurs qui peuvent être aussi modélisés par des réseaux Bayésiens. Une première étude a été menée dans [Hsa+21] pour identifier les structures de BN qui pourraient convenir. Cette étude n'est pas complètement terminée et nécessite une étude plus fine de la structure BN necéssaire pour répérer les attaques classiquement opérées sur les drones. On pourrait envisager une intégration de manière plus homogène regroupant l'aspect sûreté de fonctionnement et l'aspect sécurité en proposant des outils amont capables d'analyser les défaillances qu'elles soient intentionnelles ou pas. En effet, si l'on peut envisager une structure de BN permettant de définir un classifieur efficace d'attaques, alors le modèle réseau Bayésien pourra servir de support pour ces deux problématiques.
— la proposition d'outils d'aide à la conception de moniteurs Bayésiens adaptés à des supports variés de systèmes embarqués (FPGA, GPU, CPU). Il faut alors mettre en place des méthodes de compilation adaptés en repérant des patterns directement intégrables. Ces outils, pour être efficaces et facilement acceptables par notre communauté de chercheurs et d'ingénieurs, doivent être associés à des outils commerciaux (comme Vivado-HLS) déjà acceptés par cette communauté. Cette aide à la conception peut correspondre à des guides de conception 'intelligents' capables de faciliter la reconnaissance de patrons.
— la proposition d'outils d'estimations multi-niveaux afin d'assurer une cohérence entre la spécification fonctionnelle et la présentation opérationnelle qui respecte les contraintes de l'embarqué en terme de mémoire, de ressources de calcul, de performance et d'énergie.
— la proposition d'extensions pour une mission multi-drones. Cette proposition doit

présenter une version distribuée des moniteurs Bayésiens en prenant en compte les défaillances d'un drone ou d'une partie des drones impliqués dans le travail d'équipe.

Chapitre 4

Décision et apprentissage embarqués pour la planification de mission

Ce dernier chapitre détaille le processus de décision dans le cadre de planification de mission. Deux visions du processus de décision correspondant à des modèles différents sont plus particulièrement étudiées. La première proposition repose sur une approche multi-critères intègrant des préférences humaines. La deuxième proposition utilise des modèles probabilistes (diagramme d'influence, processus de décision de Markov-MDP) afin de mieux prendre en compte tous les aléas d'une mission.

4.1 Contexte des travaux

Dans l'approche intégrant des préférences humaines, on s'intéresse à un modèle proche de celui d'un humain en considérant les différents profils des décideurs avec des priorités variées (priorité sécurité, priorité prudence, ...). Ces travaux basés sur l'analyse des préférences humaines ont été mis en place dans la thèse d'Arwa Khannoussi en collaboration avec Patrick Meyer dans le cadre du laboratoire commun LATERAL (laboratoire commun entre le Lab-STICC et Thales LAS-OME, anciennement TOSA). Quant aux modèles probabilistes comme Markov Decision Process (MDP), ils sont classiquement utilisés dans le domaine de la robotique pour décrire une mission. Cependant ces modèles n'intègrent pas, par défaut, des problèmes de sécurité et de qualité de service. J'ai proposé dans le cadre du projet HPeC avec la thèse de Chabha Hireche d'utiliser l'approche probabiliste en prenant en compte le contexte de la mission, le système embarqué et les applications embarquées pour assurer la sûreté de fonctionnement du drone. La mise en place des alternatives associées aux différentes décisions peut être faite sur support reconfigurable. L'adaptation de la mise en oeuvre a aussi été étudiée dans le cadre de la thèse de Julien Mazuet avec la possibilité de reconfiguration dynamique qui peut offrir une solution plus économique en ressources sur FPGA, notamment dans le cadre des applications Radar.

4.2 Problématique

Considérons d'abord les éléments de décision dans le cadre de la planification de mission. Ils sont associés au contexte de la mission (éléments externes ou internes), aux alternatives (actions possibles au cours de la mission) et aux critères (poids sur certains indicateurs) pour déterminer des politiques de décision, éventuellement en fonction des profils de décideurs. Les différents modèles de décision utilisés sont évalués à partir de scénarios. Les critères de qualité des modèles reposent sur leur capacité à s'adapter au mieux au contexte applicatif, environnemental et système. Dans la proposition probabiliste à base de MDP, la difficulté est de choisir de manière judicieuse l'ensemble des paramètres du modèle, à savoir les probabilités de transition et les gains (rewards). Pour le modèle à base de préférences humaines, l'intégration des profils des décideurs est la priorité. Dans cette proposition, il faut pouvoir déterminer les profils des décideurs et définir les poids des différents critères pour la prise de décision. Pour chacune des approches, on s'intéresse à des versions embarquables des mécanismes de décisions et aussi à la mise en place de ces modèles en identifiant les méthodes d'apprentissage adaptées.

4.3 Les modèles pour la planification de mission de véhicules autonomes

Dans le cadre d'une mission de drone, on peut être amené, en fonction du contexte de la mission, à prendre des décisions. Ces décisions peuvent être prises en fonction du **contexte** de la mission et selon différents **critères** et correspondent au choix d'une **alternative** parmi un certain nombre possible. Le contexte peut être défini par différents éléments pouvant être classés en plusieurs catégories selon : a) la mission (objectifs de la mission, statut de la mission) b) la situation géographique (altitude, localisation, zone survolable, zone invisible, obstacle) c) les conditions météo (vent, pluie, nuage,...). Ces différents contextes sont identifiés par des capteurs embarqués sur le drone (GPS, altimètre, ...) ou calculés à partir de capteurs, de sondes internes ou informations transmises (par un expert externe ou cloud). Les alternatives représentent les différentes actions que le drone peut effectuer : décoller, atterrir, détecter d'une cible, suivre une cible, retourner à la base, surveiller une zone, ... Les critères qui peuvent guider le choix peuvent correspondre aux facteurs de risque (risque pour le véhicule, risque pour la mission), à la qualité intrinsèque du véhicule (énergie limitée, coût d'achat ou d'entretien), à la qualité de la mission (temps, distance, progression dans la mission). Le déclenchement des alternatives peut-être modélisé en utilisant un moteur multi-niveaux comme proposé dans [Kha+19] et illustré dans la Fig.4.1. Le premier niveau permet d'identifier si un événement qui est évalué sur de multiples critères (risque, exécution de la mission, . . .) est critique ou non. Pour le deuxième niveau, les décisions concernant le choix des actions de haut niveau

(atterrir, continuer, sauter un waypoint, . . .) sont évaluées sur de multiples critères (risque, consommation d'énergie, . . .)

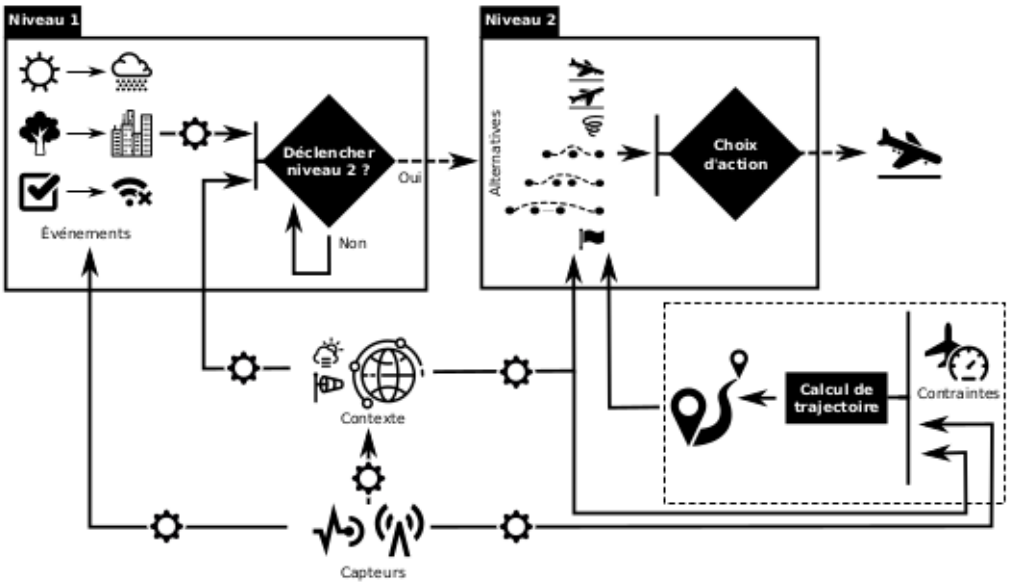

FIGURE 4.1 – Moteur de décision 3 niveaux (source [Kha19])

La prise de décision pour une planification de mission comporte différents enjeux et verrous. Ils peuvent être décrits par les points suivants :

1. Quel est le bon niveau de modélisation permettant de répondre de manière satisfaisante aux objectifs de la mission ? Identifier les éléments de contexte, les alternatives possibles et les critères utiles pour la prise de décision.
2. Quand et comment prendre en compte une décision ? Identifier les mécanismes d'adaptation/reconfigurations possibles.
3. Comment conserver une expertise humaine dans la prise de décision d'un engin totalement autonome ? Intégrer des facteurs humains dans le modèle de décision.
4. Comment gérer l'incertitude du contexte de la mission, des aléas de bon fonctionnement du véhicule ? Modéliser les différents aléas et prendre des décisions en fonctions des aléas identifiés.
5. Comment intégrer des contraintes ou des priorités dans le mécanisme de décision ?

Le premier point passe par la mise en place d'élément de langage permettant de préciser le cadre de la mission comme illustré dans la figure 4.1. Les autres points sont abordés dans les sous-sections qui suivent.

4.3.1 Adaptation/reconfiguration au cours de la mission

La prise de décision repose sur l'identification d'une situation qui permet le déclenchement d'une alternative. Puis on met en oeuvre de cette alternative en assurant la continuité

de la mission. Si l'on prend l'exemple du traitement Radar, plusieurs modes d'utilisation peuvent être envisagés : détection puis pistage, imagerie, radiocommunication, suivi de terrain. Pour chacun des modes, on peut aussi avoir des sous-modes ; pour la détection et pistage, on peut avoir les sous-modes recherche simple ou recherche avec pistage, pour l'imagerie, on peut avoir les sous-modes stripmap du radar imageur ou spotlight ou super-résolution. La difficulté est d'identifier les modes et sous-modes possibles mais aussi de savoir quand et comment on peut changer de mode.

Choisir l'action adaptée à la mission peut reposer sur plusieurs paramètres associés à

1. des évènements extérieurs comme des obstacles sur la trajectoire, un climat non favorable avec de la pluie ou un vent fort. Repérer ce type d'évènement se fait généralement via des capteurs (ex lidar pour les obstacles) ou des informations transmises par l'extérieur (via le cloud pour la météo)
2. des indicateurs de Qualité de service (QoS) pouvant renseigner en temps-réel sur la pertinence de l'application du traitement envisagé
3. des estimateurs permettant d'anticiper le paramétrage de l'application pour qu'elle puisse évoluer au moment voulu avec les bons paramètres

Dans le cas des chaînes de traitement Radar, nous avons pu identifier que les différents modes et sous-modes peuvent être liés à des métriques associées à la QoS (qualité de service). Définir ce type de métriques se fait en coopération avec les experts du domaine pour expliciter les critères de passage d'un mode (ou sous-mode) vers un autre. Par exemple pour passer du mode détection ou mode pistage, on peut se reposer sur les évaluations données par la méthode Probabilistic Data Association Filter (PDAF)[Maz+20].

Lorsque la décision de changement de mode est actée, la mise en place peut se faire de plusieurs façons en fonction des contraintes du système embarqué à bord de l'engin autonome et des contraintes liées à l'application (contraintes temps-réel, énergétiques, de ressources).

1. Par reconfiguration logicielle uniquement, le processeur du système commute d'une tâche vers une autre, en prenant soin de préserver les données si nécessaire.
2. Par reconfiguration matérielle, le GPU ou FPGA sont reparamétrés pour gérer le nouveau mode.
3. Par reconfiguration logicielle et matérielle, l'application à activer peut se déployer de manière partielle sur l'accélérateur matériel, le complément étant géré par le processeur.

L'utilisation d'un accélérateur matériel dans le processus de reconfiguration du système, que ce soit dans un contexte hybride (contexte multi-support SoC-FPGA ou SoC-GPU ou SoC-FPGA-GPU) ou contexte mono support, demande une étude plus fine du coeur de calcul (latence, ressources, énergie) et des communications avec cet accélérateur pour s'assurer d'une solution viable.

4.3.2 Intégration de facteurs humains

Le processus de décision est aussi un problème de décision multi-critères qui est guidé par le choix de l'expert ou du pilote/opérateur dans le cas d'engins non-autonomes. Dans le cas d'une mission où le drone doit survoler un ensemble de waypoints et prendre des photos pour quelques waypoints, plusieurs profils d'opérateurs peuvent être identifiés. Par exemple, un profil se focalisant essentiellement sur l'aboutissement de la mission, et un autre plus prudent qui évitera les zones trop risquées pour réaliser la mission. Pour des engins autonomes, il est possible de conserver les préférences associées aux profils des experts/opérateurs, en intégrant dans le moteur de décision ces préférences humaines. Cette aide à la décision (AMCD : Aide Multi-critère à la décision) utilisant les préférences humaines peut être présente aux différents niveaux de décision de la planification (niveaux 1,2) d'une mission comme indiqué dans la figure 4.2. Deux niveaux sont essentiellement proposés : le premier évalue les évènements qui surviennent au cours de la mission en les classant comme critiques ou non-critiques. Le deuxième niveau sert à évaluer les actions de haut niveau (atterrir, continuer, sauter un waypoint, . . .) à partir de différents critères (risque, consommation d'énergie, . . .). Ce niveau peut s'appuyer sur le module de calcul des trajectoires pour estimer certaines conséquences. Les préférences humaines modélisées définissent des profils d'expert.

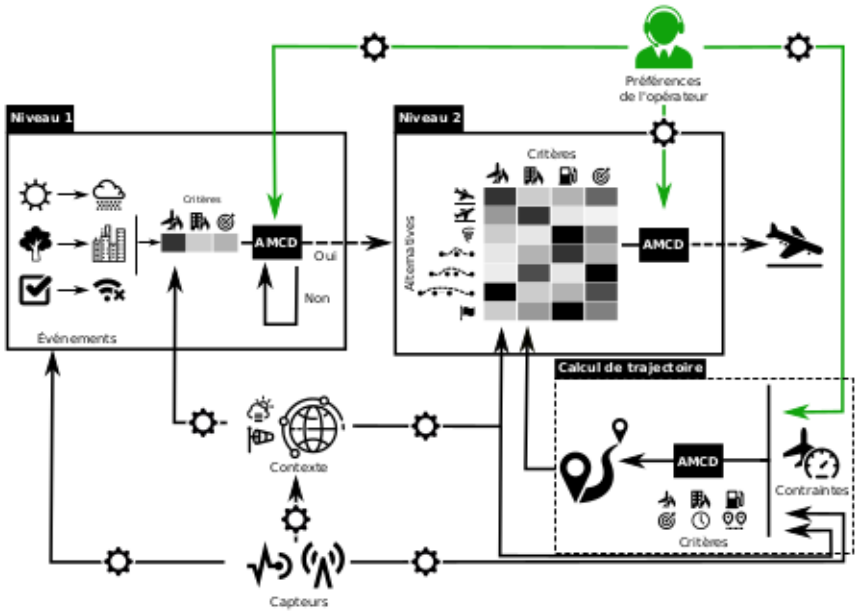

FIGURE 4.2 – Moteur de décision multi-niveaux intégrant de l'aide à la décision(source [Kha19])

Il existe différentes méthodes pour modéliser les préférences et les profils d'expert/opérateur humains dans un contexte de décision multi-critère ; par méthode de la valeur multi-attribut (MAVT) [KRM93] ou méthode de surclassement [Roy68]. Dans la thèse d'Arwa Khannoussi des méthodes de tri (MR-Sort) et de classement (SRMP) basés sur le paradigme

de surclassement ont été choisies respectivement pour la décision au niveau 1 et pour la décision au niveau 2. Pour l'aide à la décision au niveau du choix de la trajectoire, une méthode de type MAVT a été choisie. Avec la méthode Simple Ranking Method using Reference Profiles (SRMP), le classement peut s'expliquer par une série de règles, compréhensibles par l'opérateur de drone. Pour calibrer chacune des méthodes intégrant les préférences humaines, l'élicitation des paramètres doit être soigneusement étudiée. La littérature comporte de nombreux travaux sur l'élicitation des préférences de modèles MR-Sort et de modèles MAVT à fonctions de valeurs additives. Dans notre approche, l'élicitation des paramètres a été plus particulièrement étudiée pour la décision de niveau 2 pour le modèle SRMP.

4.3.3 Intégration d'aléas dans le processus de décision

Au cours des différentes missions, le véhicule autonome peut être confronté à différents types d'aléas. Ces aléas peuvent être d'ordre environnemental, dûs au climat (vent fort, pluie, perturbation électromagnétique), dûs au contexte (survol d'une ville, contexte montagneux, nuée d'oiseaux, autre drone à proximité). Ils peuvent être aussi liés à des dysfonctionnements du véhicule liés aux composants de navigation (moteur, batterie, éléments de localisation (GPS, IMU), autopilote) ou à des traitements déviants effectués par les applications embarquées, pouvant être causés par une panne sur le calculateur embarqué ou une erreur au niveau du traitement.
Ces aléas peuvent être identifiés par des composants HM (Health Monitor) représentant des moniteurs permettant de surveiller différents composants du véhicule autonome. On peut alors définir 3 grands types de composants HM :

- HM système : correspond à l'état de santé du système incluant principalement la batterie, l'autopilote, la charge du CPU, et la partie actionneur (moteurs)
- HM capteur : correspond au différents états de santé des capteurs embarqués sur le véhicule (GPS, Caméra, lidar, ...)
- HM application : correspond à la fiabilité de l'application embarquée par rapport à son contexte d'utilisation (contexte environnemental ou interne). Les applications embarquées peuvent correspondre à l'identification de cible, au suivi de cible, à la détection de zone d'atterrissage, ...

Ces moniteurs peuvent être modélisés en utilisant des réseaux Bayésiens comme on l'a montré dans le chapitre 3.

Intégrer des aléas dans le processus de décision peut se faire en se reposant sur un modèle probabiliste permettant d'associer des incertitudes à des probabilités. Dans les modèles probabilistes on trouve de manière classique les processus de Markov : MDP (markov decision process) ou des variantes avec POMDP (Partial Observable Markov Decision Process) mais aussi d'autres modèles comme les réseaux de décisions (DN), les réseaux de

Petri. Cependant d'autres modèles sont possibles pour représenter les incertitudes comme des modèles à base de logique floue, ou utilisant la théorie de Dempster-Shafer... Une vision plus complète concernant les modèles de décision est développée dans [CLN13].
Nous avons choisi d'investir des modèles probabilistes qui nous permettent d'intégrer les aléas identifiés par les différents composants HM. Deux modèles ont été plus particulièrement étudiés les réseaux de décisions (DN) et les MDP.
Le modèle DN est une extension des réseaux Bayésiens, ce qui permet une intégration très simple des HM précédemment définis. Le coeur de décision dans le cas des DN repose sur une table d'utilité affectant un score aux différentes actions en fonction des probabilités des états du système caractérisés par des BNs. Le modèle MDP permet lui une prise de décision en ayant une vision plus anticipée des conséquences de la décision prise à un moment donné. On parle, dans le cas des MDP, d'un calcul à horizon fini ou infini avec possibilité de prise en compte de manière pondérée (utilisation du facteur discount) des gains induits (reward) pour cette décision locale et pour les décisions futures possibles. Le coeur d'un MDP repose donc sur les gains associés à chaque action pour un état du système et sur les probabilités de pouvoir être dans cet état. Plusieurs méthodes de calculs peuvent être utilisées pour résoudre un MDP à horizon fini, inspirées de la programmation dynamique (algorithmes value-iteration et policy-iteration) ou des méthodes non liées à un modèle comme les méthodes de type Monte-Carlo ou de difference-temporelle (méthode Q-learning).
La résolution de la planification d'une mission de drone avec des DN ou MDPs soulève les questions suivantes :

1. Comment intégrer les différents aléas de la mission ? L'identification de ces aléas est-elle obligatoire ou peut-on s'en passer dans certains tout en préservant l'efficacité du système ?
2. Comment caractériser l'efficacité du système de décision ?
3. Comment valider le moteur de décision pour les missions de drone autonome ?
4. Comment faciliter la mise en place du modèle avec un passage à l'échelle ? et peut-on étendre le modèle à un cadre distribué ?

Pour l'intégration des aléas (question 1), nous avons proposé le modèle BFM (Bayesian network from FMEA analysis used by MDP) (voir figure 4.3). Dans ce modèle, la partie diagnostic est réalisée pour les modules HM (définis sous forme de BN). Ces modules sont mis à jour de manière continue au cours de la mission avec les données fournies par les capteurs ou par les informations de QoS fournies par les différentes applications embarquées. Le module de décision modélisé par un MDP met à jour sa décision en fonction des probabilités fournis par les modules du diagnostic pour établir le nouveau plan de mission. Les rewards permettent de définir les profils des décideurs et sont généralement stables au cours de la mission.

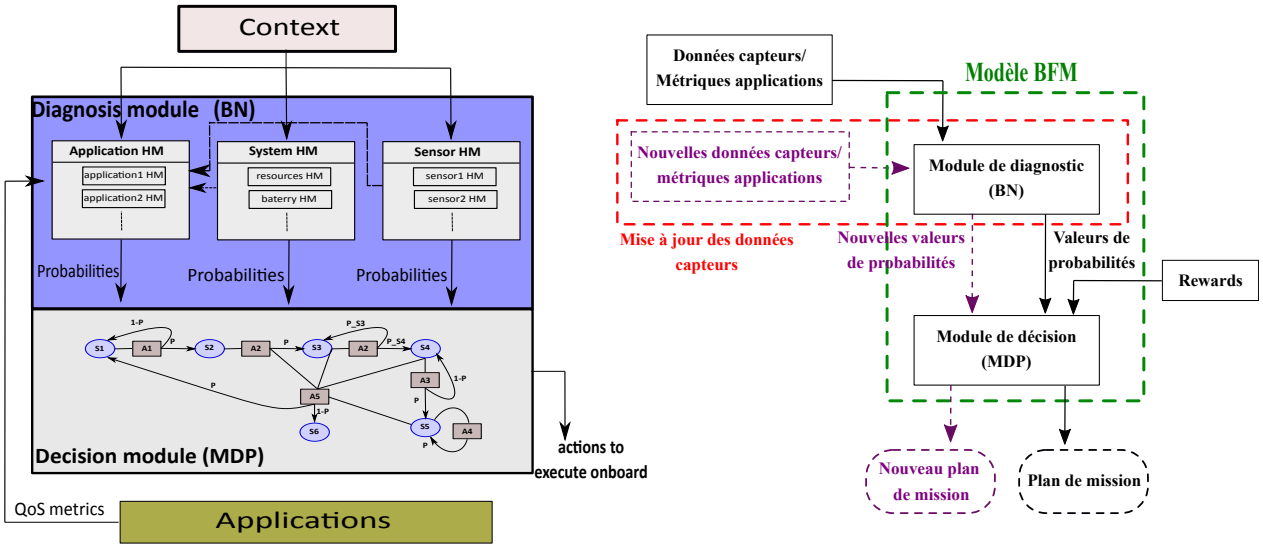

FIGURE 4.3 – Architecture du modèle BFM pour le planificateur de mission (source [Hir19])

Pour caractériser l'efficacité du modèle (question 2), nous avons utilisé essentiellement 3 métriques : la QoS, le temps d'exécution, les ressources utilisées (charge CPU, ressources FPGA, consommation d'énergie). La métrique de QoS peut être évaluée en utilisant le composant 'HM application' soumis au contexte de la mission (ex vibrations importantes détectées, vent fort, vitesse de la cible observée). Si cette valeur n'est pas satisfaisante, on peut activer des variantes des applications qui vont permettre de retrouver une valeur de QoS acceptable. La mise en oeuvre choisie pour les applications embarquées a une influence directe sur le temps d'exécution, les ressources nécessaires et la consommation d'énergie.

La validation du moteur de décision (question 3) passe par la mise en place de scénarios dans un premier temps en simulation, en activant des évènements représentant les différents contexte de l'étude. Les décisions donnent pour chaque scénario les politiques suivies sous la forme de couples Etat-Action où l'état représente l'état du système dans la modélisation MDP choisie. La cohérence des actions choisies renseigne sur la viabilité de la mission. On peut, par exemple, pour montrer l'intérêt d'une solution adaptée au contexte pour le suivi de cible, vérifier que les actions décidées permettent pour ce type de solution un meilleur suivi qu'une solution sans adaptation. Le meilleur suivi peut être défini comme un suivi plus long en termes de temps.

Pour le passage à l'échelle (question 4), il faut pouvoir faciliter la description d'une mission en proposant des décompositions plus facile à mettre en place. On a proposé de décomposer une mission complexe selon plusieurs fonctionnalités : fonctionnalité de base (applications de navigation), fonctionnalité de sécurité (détection-évitement d'obstacles, recherche zone d'atterrissage, etc), fonctionnalité applicative (applications embarquées mettant en œuvre l'objectif de la mission). Ces types de fonctionnalité peuvent être représentés par des MDPs disjoints pouvant évoluer en parallèle du moment qu'une

politique globale de gestions des conflits est définie. Définir la décision sous une forme décomposée permet d'envisager une mise à l'echelle plus facile car rajouter d'une nouvelle fonctionnalité, par exemple, peut s'envisager de manière incrémentale. Cette proposition de décomposition permet non seulement une accélération de la prise de décision mais permet aussi d'envisager un nouveau mode de coopération de MDPs distribués dans le cadre d'équipes de drones autonomes. Ce dernier aspect est actuellement à l'étude dans la thèse de Mohand Hamadouche.

4.4 Paramétrage des modèles

Une des principales difficultés des modèles de décision concerne leur paramétrage.

Ce paramétrage peut être effectué de différentes façons :

— par expertise directe ou indirecte. Les expertises sont prises en compte pour définir les seuils permettant de basculer d'une alternative vers une autre. Dans le cas du traitement Radar, le seuil pour passer d'un mode détection à un mode pistage, correspond à une métrique de QoS spécifique à ce type de traitement élaborée avec l'aide des experts. Dans le cas de décision à base de préférences humaines, l'expertise des opérateurs est transmise de manière indirecte par le biais de questionnaires.

— par déduction en analysant les historiques des actions passées. L'historique des décisions passées peut servir de base pour définir le paramétrage du modèle. Il faut généralement que ce jeu de données associé aux historiques des décisions soit alimenté avec un nombre représentatif d'expériences.

— par renforcement en effectuant des essais virtuels successifs. Avant la phase de décision, on a une phase d'exploration qui permet d'adapter les décisions au problème à résoudre avant de passer à la phase exploitation. Cette phase d'exploration peut être vue comme un paramétrage du système et est notamment utilisée dans les méthodes de type Q-learning.

Les éléments constituant le paramétrage dépendent du modèle utilisé. Ces éléments sont détaillées pour quelques modèles de décisions dans les paragraphes qui suivent.

4.4.1 Elicitation de paramètres pour un modèle intégrant des préférences humaines

Pour le modèle SRMP, on doit identifier les paramètres de préférences du décideur/opérateur en prenant en compte les différents profils de références, l'ordre lexicographique des profils de références puis les poids des critères utilisés. Les préférences du décideur sont exprimées sous la forme de préférences sur des paires d'alternatives puis analysées par des algorithmes d'élicitation proposant une formulation exacte du problème (ex : la programmation linéaire mixte en nombres entiers : MIP ou MILP) ou

une formulation approchée (ex : algorithme génétique). Ce programme est construit à partir d'une liste de comparaisons binaires indiquées par le décideur. Cette liste contient des paires de préférence stricte ou d'indifférence, qui génèrent des contraintes pour le MIP. Avec un algorithme génétique, différentes configurations et différents opérateurs sont testés pour trouver les mieux adaptés pour résoudre notre problème et des tests supplémentaires doivent être mis en place pour évaluer le comportement de notre algorithme face à de nouvelles alternatives.

La phase d'élicitation des préférences se passe en amont de la mission. Comme cette phase peut nécessiter un nombre élevé de comparaisons binaires, cela peut d'engendrer un effort cognitif important pour l'opérateur (et qui risque par conséquent d'émettre des préférences non-compatibles avec le modèle élicité). Pour répondre à ce problème de fatigue, une approche incrémentale pour l'élicitation des paramètres est proposée dans [Kha+18a]. Dans l'approche incrémentale, on peut choisir entre deux types de solutions : la solution exacte à base de MIP ou la solution approchée à base d'heuristique avec un algorithme génétique (AG). La version AG offre une solution moins coûteuse en temps cependant les résultats restent moins performants que dans le cas MIP. Pour obtenir des résultats similaires entre les deux approches, il faut rajouter un ensemble complémentaire de paires d'apprentissage dans la version AG.

4.4.2 Apprentissage des différents modèles intégrant des aléas

Dans les modèle de décisions probabilistes tel que les MDPs, les paramètres du système concernent les probabilités de transitions et les rewards associées aux actions. Ces deux informations peuvent être prises en compte de la manière suivante :

1. Pour les probabilités concernant les transitions : on a plusieurs solutions
   (a) valeur donnée par un expert
   (b) valeur estimée : une evaluation est effectuée basée sur la base d'observation et d'indicateurs internes. C'est le cas lorsque l'on utilise des HM à base de BN. Les probabilités sont calculées en fonction des observations faites et des corrélations mises en place grâce à la structure du BN.
   (c) valeur ignorée : l'information peut être manquante. On peut alors utiliser une variante du modèle MDP comme le Q-learning qui peut trouver la meilleure solution après une phase d'exploration. L'efficacité de cette approche a été étudiée dans [Ham+21] et donne des résultats intéressants. En effet, même si l'on peut naturellement penser qu'un modèle complet donne toujours des meilleurs résultats, on peut montrer que dans des cas simple et réguliers/homogènes, le Q-learning se montre tout à fait efficace en termes de qualité du résultat et temps d'exécution. Cependant pour les cas plus complexes comme les missions

de drones, les informations concernant les probabilités sont indispensables pour converger vers un résultat correct en des 'temps acceptables'.

2. Pour les rewards : on a recours généralement à des méthodes empiriques et trouver les bonnes valeurs se fait souvent par essais successifs avec des valeurs prises de manière aléatoires. Cependant évaluer ces valeurs des rewards est utile dans le cadre de la mission de drone :
   (a) pour définir un profil de mission : profil sécurité, profil mission. Seules les valeurs des rewards sont ajustées pour assurer ces profils.
   (b) pour assurer une cohérence entre différents MDPs et pour définir des niveaux de priorités entre actions se trouvant réalisées par différents MDPs ou définir des contraintes entre actions ou actions-évènements. Ce point à été développé dans [HDB20] et est actuellement la base des travaux sur la mise en place d'une décision distribuée.

4.5 Version embarquée de la décision

Dans le cas des véhicules autonomes, les décisions sont nécessairement embarquées. Il faut que le moteur de décision puisse statuer en fonction des évènements, des aléas et assurer la mise en place d'alternatives. Nous nous sommes intéressés à ces deux points dans une version embarquée de la décision. Nous présentons d'abord la mise en place des alternatives qui peut s'appuyer sur le mécanisme de reconfiguration pour pouvoir adapter l'action ou le traitement. Puis nous nous intéressons au coeur du moteur de décision et à la pertinence de définir un accélérateur matériel pour réaliser le calcul d'inférence ou pour l'apprentissage du modèle afin de caractériser les différents paramètres.

4.5.1 Mise en place de la reconfiguration du système

La mise en place de reconfiguration dans le cadre matériel ou mixte (logiciel/matériel) passe par l'identification des alternatives avec le détail de leurs variantes en termes d'architectures. La reconfiguation matérielle s'appuie sur les mécanismes de reconfiguration du FPGA (globale ou partielle, statique ou dynamique). Si la reconfiguration est partielle, elle nécessite alors l'identification de la zone à reconfigurer en termes de ressources matérielles, l'adaptation des entrées et sorties de cette zone à la nouvelle IP et l'évaluation en temps d'exécution de l'utilisation de cette nouvelle variante incluant le temps propre à la reconfiguration (temps de chargement du bitstream associé).

Dans le contexte des applications Radar, le traitement DOPPLER utilise classiquement des DFT et FFT. La FFT est préconisée pour la recherche de la cible tandis que la DFT est plus efficace dans le cadre du pistage. Le passage de la DFT à la FFT et inversement dépend du contexte de la mission et si la détection de la cible est faite ou non comme illustré

dans la figure 4.4. En termes d'architectures, les deux solutions offrent des caractéristiques différentes : en latence (latence de $9,3$ ms pour la FFT contre une latence $6,6$ ms pour la DFT), en ressource (FFT occupe en moyenne 3 fois plus de ressources sur le FPGA que la DFT), nombre de traitements d'échantillons (100×125 échantillons pour FFT contre 32×32 échantillons pour la DFT) avec une initialisation de calculs différents pour chacune des méthodes. Dans le cadre du traitement Radar, le moteur de décision utilise une métrique de QoS qui est calculée en logiciel. La mise en oeuvre de la reconfiguration dynamique passe alors par la détection de seuil qui assure la transition d'un mode à l'autre. Ce mécanisme de détection est effectuée uniquement sous forme logiciel dans ce contexte Radar et déclenche la reconfiguration matérielle d'une zone de FPGA.

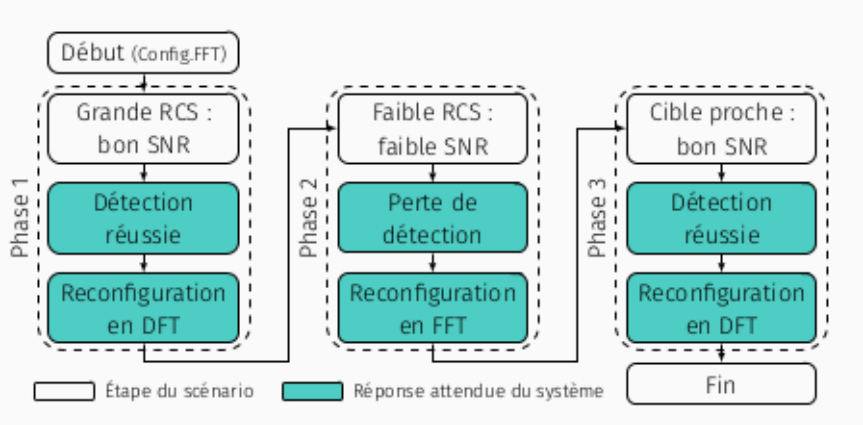

FIGURE 4.4 – Reconfiguration au cours d'une mission de détection/pistage de cible effectuée par traitement DOPPLER (source [Maz21])

4.5.2 Accélérateur matériel pour le calcul d'inférence du moteur de décision

Dans cette partie, on essaie de répondre aux questions suivantes : est-il intéressant de définir des accélérateurs matériels pour les moteurs de décision ? Peut-on espérer des accélérations pour l'inférence ?

En effet, il est classique d'entraîner le modèle hors ligne et de générer notamment dans le cas des MDPs la table des actions qui est utilisée pour une exécution en ligne. Ce raccourci est possible si les paramètres des modèles restent stables au cours de la mission. Cependant ce type de raccourci ne peut pas être envisagé si les paramètres évoluent de manière dynamique. Dans ce cas, le modèle complet doit être embarqué. L'intérêt concernant l'accélération matérielle des moteurs de décision dépend du modèle utilisé, de sa complexité et des pré-traitements nécessaires pour une utilisation des plus optimales. Nous avons effectué quelques études d'accélérateurs matériels pour les moteurs de décision spécifiques (partie inférence) :

— à base d'arbre de décision [KDM16]

Modèle inférence	Cas de figure	Accélération sur Zynq (Temps SW/ Temps HW)	Frein à l'accélération
Réseaux de décision	[Zer17]	1 à 21, 3	Accès aux données
MDP	[Hir19]	1 à 2, 86	Peu de parallélisme
CNN (LeNet)	[Fel20]	24, 43	Volume de données

TABLE 4.1 – Quelques exemples d'accélérateurs matériels étudiés pour le calcul d'inférence sur Zynq

— à base de réseaux de décision [DZH20]
— à base de MDP [Hir19]
— à base de réseaux de neurones [Le +20][Fel20]

Il ressort de ces études les points suivants :

1. les pré-traitements peuvent permettre de réduire considérable le coeur de décision. Par exemple dans l'étude de [Liu+20], le pruning dans les CNN peut réduire jusqu'à 95% les ressources et jusqu'à 80% le temps d'exécution sans trop modifier la qualité du modèle (perte de précision inférieure à 4%).
2. le stockage et l'accès aux paramètres des modèles sont des éléments critiques pour l'amélioration des performances et souvent sources de compromis entre adaptivité du modèle et performance. En effet, le modèle le plus efficace est généralement celui qui intègre les paramètres dans la solution matérielle comme des constantes permettant ainsi des simplifications du modèle mis en place (possibilité d'optimisation avec des propagations des constantes associées aux paramètres). Le typage des données utilisées permet aussi de réduire la quantité de ressources (FPGA) à utiliser notamment en passant du type flottant à la virgule fixe avec une largeur ajustée au mieux pour conserver la qualités des traitements.
3. Le coeur de calcul des modèles peut être relativement 'simple' car il peut être parfois réduit à quelques lignes de code. C'est le cas des arbres de décision, des réseaux de décision, et des MDPs. La principale difficulté est la réécriture des calculs sous forme non récursive afin qu'ils puissent être synthétisables en utilisant les outils de HLS.

A titre d'exemple, le tableau 4.1 donne des valeurs d'accélérations pour le calcul d'inférence en utilisant quelques uns de ces modèles. Les accélérations affichées ont été obtenues sur le processeur Zynq de la carte Zedboard en comparant les temps d'exécution sur ARM et les temps d'exécution sur la partie reconfigurable. Cependant, quelques propriétés inhérentes aux modèles sont identifiées comme un frein pour l'accélération. Ces freins concernent notamment la séquentialité des algorithmes et/ou l'accès aux données.

Dans les réflexions menées, il reste des cas pour lesquels l'élaboration d'accélérateurs matériels n'est pas pour l'instant souhaitable. Ils concernent les moteurs de décisions à base de solveurs MILP qui font appel des heuristiques complexes et donc difficiles à

réexprimer de manière simple. Le travail de reprise des heuristiques nous a semblé un travail trop volumineux. De plus, ces heuristiques utilisent beaucoup de récursivité difficile à gérer pour les outils de HLS commerciaux sans réecriture vers une version itérative. Par ailleurs, il existe actuellement des versions de ces solveurs mathématiques très efficaces sur machines serveurs ou distribuées rendant la concurrence avec une mise en oeuvre dédiée sur FPGA plus difficile à envisager.

Cependant, il reste une voie alternative pour des accélérateurs matériels dans le cadre de recherche de solutions multi-objectifs qui est de passer par des solutions approchées utilisant des méta-heuristiques. En effet, les algorithmes évolutionnaires (ou génétiques) sont de bons candidats pour définir une solution embarquable pour ce type de problème.

En conclusion pour la partie inférence, le coeur du moteur de décision offre quelques possibilités d'accélérations sur support matériel reconfigurable fortement dépendantes du modèle sous-jacent utilisé. Dans de nombreux cas, une solution sur CPU permet de répondre aux contraintes temps-réel de l'application et de la mission. Cependant, concevoir des solutions matérielles pour le moteur d'inférence reste toute fois une alternative intéressante par rapport à une version logicielle dans les cas de figure suivants :

1. afin de pouvoir décharger le processeur de l'activité de décision
2. afin pouvoir migrer le coeur de décision au plus près des estimateurs qui sont les sources d'alimentation de ce moteur. Donc de permettre un couplage diagnostic-décision fort au niveau FPGA, au plus proche des capteurs, et sans retard pouvant être dus aux transferts entre CPU et FPGA. Nous avons exploré ce point dans les modèles DN et BFM et les conclusions de l'article [DZH20] vont dans ce sens.

4.5.3 Un accélérateur matériel pour de l'apprentissage

Après avoir traité le cas du calcul d'inférence pour une mise en oeuvre matérielle, on considère à présent le cas de l'apprentissage. Généralement l'apprentissage se fait sur un serveur de grande capacité ou des réseaux de machines distribuées, il est actuellement difficile d'imaginer embarquer ce type de calcul dans le cas général. Cependant, on peut envisager le raffinement du modèle existant pour des paramètres ciblés. Ces paramètres peuvent être appris (apprentissage partiel ou incrémental) ou peuvent être transmis par une autorité extérieure.

1. Dans le cas d'un apprentissage incrémental, le mécanisme d'apprentissage peut se trouver allégé et une mise en oeuvre en ligne peut être possible en fonction du temps d'apprentissage. Par exemple, dans le cas de drone comme le Watchkeeper avec une autonomie de plusieurs heures de vol, ce type de traitement peut être envisageable.

2. Pour le transfert des paramètres du modèle par une autorité extérieure, il est possible d'utiliser des techniques de type federated learning [Li+20] pour la mise en oeuvre.

Dans le cadre de l'apprentissage en ligne, on s'est intéressé aux schémas partiels ou incrémentaux de l'apprentissage pour les cas suivants :

— Dans le cas de l'élicitation incrémentale du modèle SRMP intégrant des facteurs humains avec une approche heuristique. La version embarquée de ce type d'apprentissage repose sur la mise en oeuvre matérielle d'un algorithme génétique. Même si cette étude, abordée lors d'un stage de master, n'a pas pu aboutir complètement, elle laisse cependant espérer des accélérations intéressantes.

— Dans le cas du raffinement du modèle BFM, deux types de raffinements sont possibles pour le MDP sous-jacent ; au niveau des BNs qui calculent les valeurs des transitions ou au niveau des rewards du MDP. Dans le premier cas, c'est l'apprentissage du réseau BN qui est effectué (cas déjà évoqué dans le chapitre précédent) pour mieux évaluer les valeurs de transition. Pour les rewards, on peut être amené à revoir leurs valeurs en fonction des stratégies de mission ciblées (priorité sur la sécurité ou priorité sur la mission) [Hir19]. Ces rewards peuvent aussi s'adapter à différentes contraintes de l'environnement et être recalculés comme proposé dans [HDB20].

Actuellement, le transfert de paramètres via les techniques de federated learning n'a pas été encore étudié pour ces modèles mais pourrait être tout à fait envisageable dans un cadre de décision distribuée.

4.6 Conclusion sur la décision et apprentissage

Dans ce chapitre, nous avons étudié les mécanismes de décision dédiés à la planification de missions. Différents types de modèles ont été abordés permettant l'intégration des préférences humaines et la prise en compte d'aléas. La partie embarquée de ces mécanismes correspond essentiellement au calcul d'inférence de la décision pouvant éventuellement être accélérée par des mises en oeuvre matérielles sur FPGA. Des versions embarquées peuvent aussi utiliser des mécanismes de reconfiguration (statique ou dynamique) du FPGA pour optimiser les ressources de celui-ci. La caractérisation des modèles de décision se fait en général hors ligne en utilisant des mécanismes d'apprentissage. Elle peut, en effet, prendre des heures voire quelques jours d'apprentissage avant de déterminer les meilleurs modèles. L'apprentissage en ligne reste alors actuellement peu envisageable dans le cadre d'une mission de drone (qui dure en général quelques minutes pour des petits UAV de type quadrirotors). Cependant, on peut envisager un apprentissage incrémental ou partiel afin de raffiner un modèle déjà en place. Dans ce dernier cas, les accélérateurs matériels ont toute leur place.

4.7 Perspectives sur la décision pour des missions de drones

Deux types de perspectives peuvent être proposées dans le cadre de cet axe.

La première concerne le raffinement du modèle utilisé pour la prise de décision qui peut être fait 'at the edge'. Ce processus de raffinement peut venir en appui du processus d'apprentissage plus grossier du modèle pouvant être fait en amont. Avec l'intégration de super-calculateur dans les systèmes embarqués à plus long terme, on pourra aussi envisager de refaire complètement l'apprentissage des modèles. On voit déjà actuellement des propositions dans ce sens [Zij+21] utilisant des modèles de type Deep Reinforcement Learning (DRL).

La deuxième perspective s'inscrit dans le cadre de la décision distribuée reposant sur le mécanisme de coopération au sein d'une équipe ou d'un essaim de drones. Dans cette extension, on peut étudier :

- Le déploiement d'un service ou d'une mission sur un ensemble de ressources autonomes. Ce problème peut s'apparenter à un problème d'allocation de tâches sur un réseau de calculateurs. Ce réseau pouvant contenir des éléments défaillants.
- La mise en place de la sécurité d'un service se reposant sur ensemble de ressources autonomes et non d'un véhicule uniquement
- La mise à jour de la décision distribuée et coordonnée. Les modèles de décision embarquée peuvent avoir besoin d'une mise à jour. Le calcul de cette mise à jour peut s'imaginer en ligne ou hors ligne. La mise à jour effective doit être optimisée de manière cohérente par rapport à l'ensemble des entités de l'équipe ou de l'essaim impliquées.

Chapitre 5

Perspectives de recherche

Dans ces travaux de recherche, les activités autour de l'IA embarquée ont été évoquées. Elles concernent plus particulièrement les modèles pouvant assurer la sûreté de fonctionnement et la prise de décision dans le cadre des drones autonomes. Elles s'inscrivent dans une ligne directrice plus générale de méthodologie de conception d'accélérateurs matériels pour l'embarqué. Quelques perspectives à ces travaux sont présentées dans les sections qui suivent. On présente d'abord les perspectives dans les grandes lignes avant d'identifier les actions possibles à court et long terme.

5.1 Evolutions des Axes de recherche

Les perspectives de recherche envisagées viennent renforcer de manière préférentielle les deux derniers axes de recherche et élargissent la thématique au domaine de la sécurité et de l'IA distribuée et embarquée.
Dans ces perspectives, on trouve les thématiques suivantes qui correspondent à des travaux déjà initialisés qui nous paraissent mériter des approfondissements :

1. les méthodes d'apprentissage embarquées intégrant des aspects contextuels[Le +20]
2. les méthodes de décision embarquée pour IoT et pour essaims d'IoTs intégrant des services fiables [Bra20] [HDB20]
3. l'identification des attaques cyber pour renforcer la sécurité des véhicules [Hsa+21].

Le premier point correspond à des travaux de recherche sur du court terme avec le projet VISEMAR. Les deuxième et troisième points correspondent à des travaux avec l'USP du Brésil (Projet NORAH et plus récemment le projet STRAUSS) sur des services sécurisés à déployer au sein d'une équipe de drones. Dans le cadre d'une réduction de consommation d'énergie, on souhaite s'intéresser au edge computing pour assurer une partie des services. Les travaux en perspectives peuvent s'articuler comme suit avec les axes présentés dans ce manuscrit :

— Pour l'Axe 1 : Evoluer vers une méthodologie de conception intégrant des méthodes d'ingéniérie (méthodes agiles) pour la reconfiguration dynamique ou pour la gestion d'implémentations multi-supports avec des accélérateurs dédiées (par exemple : Deep learning sur support hybrides FPGA, GPU, Multi-coeurs).

— pour l'Axe 2 : Augmenter les FDIR probabilistes dédiés au diagnostic avec des capacités de détection d'attaques de sécurité et pouvant proposer des contre-mesures, et des mises à jour en ligne de modèle.
— pour l'Axe 3 : Etendre les mécanismes de décision à une décision distribuée au sein d'une équipe de véhicule autonomes équipés de plateformes hétérogènes. Favoriser le calcul local 'at the edge', et mettre en place des techniques d'apprentissage en ligne pour une mise à jour continuelle du modèle.

5.2 Perspectives à court terme

1. Pour le projet VISEMAR, la thèse de Tanguy Le Pennec démarrée en novembre 2018 dans le cadre du labo commun WAVES (Lab-STICC/ISEN/Thales) permet de renforcer les activités sur l'apprentissage et la reconnaissance embarquées d'objets spécifiques se trouvant dans les fonds marins. Dans cette thèse, des modèles de décision et d'apprentissage de type Deep Learning seront abordés pour effectuer de la reconnaissance dans des fonds marins éventuellement perturbés. Les travaux portent essentiellement sur la mise en place de l'architecture du réseau dédié à la reconnaissance d'objets ciblés (5 classes ont été retenues et correspondent à un fond, un rocher, un poisson, une plante et du sable) et à la mise en place d'une chaine d'outils pour une conception facilitée sur FPGA.
2. Pour le projet NORAH : la thèse de Mohand Hamadouche démarrée en octobre 2019 permet d'investir le cadre de modèles de décision distribuée pour assurer la sécurité des services opérés par des drones autonomes. Dans ce projet, nous proposons une surveillance permanente des attaques ou défaillances possibles au niveau de l'essaim de drones ou d'IoTs (Internet of Things), en mettant en place des moniteurs intelligents capables d'estimer, en fonction du contexte de la mission, l'état du système embarqué, des applications embarquées et de l'état des communications entre drones. Ces dispositifs seront intégrés à un moteur de prise de décision du drone. Ils pourront être élaborés à partir de modèles probabilistes en considérant de manière attentive le cadre des menaces aériennes et le cadre distribué des essaims de drones. Dans ce projet, nous souhaitons étendre le cadre de la sûreté de fonctionnement précédemment étudié avec les propositions suivantes : 1) proposition d'estimation des menaces de sécurité et d'élaboration de contre-mesures en suivant un formalisme commun aux techniques de sûreté de fonctionnement, 2) proposition d'intégration des estimateurs dans le processus de décision d'un engin individuel ou d'un engin coopérant au niveau d'un essaim de drones, 3) mise à jour des estimations en ligne pour une revisite des paramètres du modèle si besoin. Plusieurs enjeux scientifiques sont ici identifiés : 1) utiliser un même modèle probabiliste pour élaborer un estimateur des risques de défaillance/sécurité 2) définir une

méthode/outil informatique permettant de faciliter la spécification, la génération (version software ou hardware) de ces estimateurs et leur intégration au niveau du drone ou de l'essaim 3) puis valider les propositions par des mises en situation de mission et explorer la possibilité de mettre à jour modèle déployé sur les drones si nécessaire. De plus, pour répondre aux contraintes temps-réel et d'énergie d'une utilisation en ligne, des implémentations variées des moniteurs/estimateurs peuvent être proposées sur différents supports (CPU, GPU, FPGA).

5.3 Perspectives à moyen et long termes

Plusieurs perspectives sont envisagées à moyen et long termes et concernent :

1. La poursuite des activités autour de la **sûreté de fonctionnement et sécurité dans le processus de décision**. On envisage de renforcer de la dimension cybersécurité pour faire face aux attaques cyber pour des drones aériens ou réseaux acoustiques (projet STRAUSS, projet Cormorant, usine du futur).

   Les actions actuellement entreprises pour ce type d'activités se déclinent selon ces trois contextes pouvant déboucher sur des travaux de thèse :

   (a) le projet STRAUSS dans le contexte des ville intelligentes (dépôt de projet ANR PRCI en 2022) se focalise sur les services sécurisés pour la mise en place de taxis volants : thèse STRAUSS

   (b) le projet Netoptim (GIS Cormorant) initialement sur la sécurité des réseaux acoustiques, devient un projet de mise en place d'IDS (Intrusion Détection System) génériques pour réseaux aériens et acoustiques : thèse Cormorant/Thalès

   (c) le projet Usine du futur doit traiter de la coopération entre drones hétérogènes (aérien/terrestre). Les drones viennent en soutien dans la gestion des défaillances d'une usine 4.0. permettant de simplifier de processus de détection de pannes sans intervention humaine.

2. Le renforcement de la thématique outil et méthode pour **IA embarquée reposant sur une approche intégrée** avec les thèses démarrées à l'automne 2018/19. Dans cette thématique, on souhaite porter des techniques IA sur des systèmes embarqués. Le but dans cette approche est de proposer une spécialisation, du 'sur-mesure' des techniques IA généralement effectuées sur de gros serveurs de calculs afin de réduire les ressources nécessaires. Dans le contexte de surveillance sous marine par exemple, l'intégration des contextes spécifiques d'utilisation comme la turbidité des fonds marins permet d'envisager des techniques 'Learning in the field' à base de Transfert Learning utilisables dans le cadre embarqué.

   (a) eHPC : calculateur embarqué Haute Performance en respectant les contraintes ressource/latence/energie. L'IA est un grande consommatrice de calculs, de

mémoire. Ces besoins sont similaires à ceux du HPC, avec des optimisations énergétiques qui sont de plus en plus critiques et conditionnant leur embarcabilité.

(b) IP IA modulable et adaptable au contexte et plateforme cible (FPGA, GPU, CPU) : mise à disposition de librairies de Modèle IA précompilés pour éviter des temps de synthèse ou des temps de mises en place sur le support matériel trop longs.

(c) IA dans le processus de conception de nouvelles IP : participer à la recommandation de fonctionnalité pour sécuriser (utilisation de patrons pour faciliter l'aide à la conception), pour faciliter le raffinement continu (redimensionner les éléments de solution, raffiner les paramètres avec un processus incrémental), pour valider et tester de manière incrémentale (métriques utilisées en continu au cours du processus pour valider les différentes solutions). La thèse de Perig Dissaux (démarrée en février 2022) s'inscrit dans cette thématique.

Bibliographie

Références associées aux activités de recherche

[All+18] Malou Alleno et al., *Éduquer en montrant l'exemple : les filles qui... (descriptif de poster)*, Didapro 7 – DidaSTIC. De 0 à 1 ou l'heure de l'informatique à l'école, Poster, fév. 2018, URL : https://hal.archives-ouvertes.fr/hal-01753227.

[And+05a] Caaliph Andriamisaina et al., « Abstract Synthesis of Turbo Decoder Elements onto Reconfigurable Circuit », *in* : *ERSA (International Conference in Engineering of Reconfigurable Systems and Algorithms)*, Las Vegas, USA, juin 2005, URL : https://hal.archives-ouvertes.fr/hal-00083398.

[And+05b] Caaliph Andriamisaina et al., « Synthèse abstraite d'éléments de turbo-décodeurs en bloc pour circuits reconfigurables », *in* : *SympA'2005 : 8ème édition du symposium en architectures nouvelles de machines*, France, 2005, p. 25–36, URL : https://hal.archives-ouvertes.fr/hal-00083359.

[Bra20] Isadora Ferrão Daniel Pigatto João Fontes Natassya Silva David Espes Catherine Dezan Kalinka Branco, « STUART : ReSilient archiTecture to dynamically manage Unmanned aeriAl vehicle networks undeR atTack », *in* : *8th Workshop on Communications in Critical Embedded Systems(IEEE ISCC/WoCCES)*, Rennes, France, juil. 2020.

[CA02] Dezan Catherine et Plantec Alain, *Simulation distribuée de systèmes VHDL : étude de cas, système à base de ST3500*, Rapport de contrat, UBO, oct. 2002.

[Dez+06a] Catherine Dezan et al., *Rapport et bilan pour le projet VALMADEO pour étape 1*, Rapport de contrat région (projet PRIR), mar. 2006, URL : http://hal.univ-brest.fr/hal-00487924.

[Dez+06b] Catherine Dezan et al., « Synthèse portable pour micro-architectures à grain fin. Application aux turbo-décodeurs et nano-fabriques. », *in* : *Revue des Sciences et Technologies de l'Information - Série TSI : Technique et Science Informatiques* 25 (2006), p. 893–920, URL : https://hal.archives-ouvertes.fr/hal-00083721.

[Dez+06c] Catherine Dezan et al., « The Case Study of Block turbo Decoders on a Framework for Portable Synthesis on FPGA », *in* : *39th Hawaii International conference on System Sciences*, United States : IEEE Computer Society, 2006, 250b, URL : https://hal.archives-ouvertes.fr/hal-00083395.

[Dez+07] Catherine Dezan et al., « Building CAD Prototyping Tool for Emerging Nanoscale Fabrics », *in* : *ENS 2007*, Submitted on behalf of EDA Publishing Association (http ://irevues.inist.fr/handle/2042/5920), Paris, France : EDA Publishing, déc. 2007, p. 25–30, URL : https://hal.archives-ouvertes.fr/hal-00202507.

[Dez+08] Catherine Dezan et al., *Rapport et bilan pour le projet VALMADEO pour l'étape 3*, Rapport contrat région (projet PRIR), nov. 2008, URL : http://hal.univ-brest.fr/hal-00487350.

[Dez+09] Catherine Dezan et al., « Towards a Framework for Designing Applications onto hybrid nano/CMOS fabrics », *in* : *Microelectronics Journal* 40.*4-5* (avr. 2009), p. 656–664, DOI : 10.1016/j.mejo.2008.07.072, URL : http://hal.univ-brest.fr/hal-00379170.

[Dez+91] C. Dezan et al., « Synthesis of systolic arrays by equation transformations », *in* : *International Conference on Application Specific Array Processors*, Barcelona, France : IEEE Comput. Soc. Press, sept. 1991, DOI : 10.1109/ASAP.1991.238911, URL : http://hal.univ-brest.fr/hal-01862745.

[Dez+92] Catherine Dezan et al., « The Alpha Du Centaur Experiment », *in* : *Proceedings of the International Workshop on Algorithms and Parallel VLSI Architectures II*, Gers, France : Elsevier Science Publishers B. V., 1992, p. 325–334, ISBN : 0-444-89153-6, URL : http://dl.acm.org/citation.cfm?id=146350.146534.

[Dez12] Catherine Dezan, « From Specifications towards Hardware », *in* : *Invited seminar, ARCAA*, Conférence invitée ARCAA (Australian Research Centre for Aerospace Automation), Brisbane, Australia, juil. 2012, URL : http://hal.univ-brest.fr/hal-00765035.

[Dez15] Catherine Dezan, « Embedded Health Management for Autonomous UAV Mission », *in* : *QUT Robotics seminar*, Brisbane, Australia, nov. 2015, URL : http://hal.univ-brest.fr/hal-01443248.

[Dez16] Catherine Dezan, « Embedded Diagnosis and Mission Planning based on Stochastic Methods », *in* : *USP ICMC Robotics seminar*, San Carlos, Brazil, oct. 2016, URL : http://hal.univ-brest.fr/hal-01443249.

[Dez18a] Catherine Dezan, « Bayesian Networks for Safety/Security (USP seminar) », working paper or preprint, fév. 2018, URL : http://hal.univ-brest.fr/hal-01843694.

[Dez18b] Catherine Dezan, « IA mission planning for autonomous vehicles : probabilistic models and embedded versions », *in* : *IX Escola Regional de Informatica SP/Oeste*, San Carlos, Brazil, mar. 2018, URL : http://hal.univ-brest.fr/hal-01843697.

[Dez93] Catherine Dezan, « Géneration automatique de circuits avec ALPHA du CENTAUR », thèse de doct., Rennes 1, 1993.

[Dez94] C. Dezan, « Generating Regular Arrays By Program Transformations », *in* : *Second Euromicro Workshop on Parallel and Distributed Processing*, Malaga, France : IEEE, jan. 1994, DOI : 10.1109/EMPDP.1994.592486, URL : http://hal.univ-brest.fr/hal-01862751.

[DGQ90] Catherine Dezan, Eric Gautrin et Patrice Quinton, *Conception et intégration d'un correlateur systolique*, Research Report RR-1351, INRIA, 1990, URL : https://hal.inria.fr/inria-00075208.

[DGQ91] Catherine Dezan, Eric Gautrin et Patrice Quinton, « Conception et intégration d'un corrélateur systolique », *in* : *Annales des Télécommunications* 46 (jan. 1991), p. 69–77, URL : https://hal.archives-ouvertes.fr/hal-01863704.

[DLP99] C. Dezan, L. Lagadec et B. Pottier, « Object oriented approach for modeling digital circuits », *in* : *1999 IEEE International Conference on Microelectronics Systems Education (MSE'99)*, Arlington, United States : IEEE Comput. Soc, juil. 1999, DOI : 10.1109/MSE.1999.787033, URL : http://hal.univ-brest.fr/hal-01862761.

[DQ94a] C. Dezan et P. Quinton, « Verification of regular architectures using ALPHA : a case study », *in* : *IEEE International Conference on Application Specific Array Processors (ASAP'94)*, San Francisco, France : IEEE Comput. Soc. Press, août 1994, DOI : 10.1109/ASAP.1994.331806, URL : http://hal.univ-brest.fr/hal-01862756.

[DQ94b] Catherine Dezan et Patrice Quinton, *Verification of regular architectures using ALPHA : a case study*, Research Report RR-2284, INRIA, 1994, URL : https://hal.inria.fr/inria-00074388.

[DW07] Catherine Dezan et Teng Wang, « Introduction of Error Correcting Schemes in the design Process of Self-Healing Circuits for Nanoscale Fabrics », *in* : *Eleventh Annual HPEC workshop : High Performance Embedded Computing*,

Lexington, United States : CSREA Press, sept. 2007, p. 25, URL : https://hal.archives-ouvertes.fr/hal-00169921.

[DZH20] Catherine DEZAN, Sara ZERMANI et Chabha HIRECHE, « Embedded Bayesian Network Contribution for a Safe Mission Planning of Autonomous Vehicles », *in* : *Algorithms* 13.*7 :155* (2020), DOI : 10.3390/a13070155, URL : https://doi.org/10.3390/a13070155.

[Fra+18] Matheus FRANCO et al., « Model-Based Dependability Analysis of UnmannedAerial Vehicles - A Case Study », *in* : *48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (SSIV'18)*, Luxembourg, Luxembourg, juin 2018, URL : https://hal.univ-brest.fr/hal-01843133.

[Gou+07] Thierry GOUBIER et al., *Projet VALMADEO, étape 2*, Rapport de contrat région (projet PRIR), juin 2007, URL : http://hal.univ-brest.fr/hal-00487915.

[Gou+08] Thierry GOUBIER et al., « Fine Grain Parallel Decoding of Turbo Product Codes : Algorithm and Architecture », *in* : *5th international symposium on turbo codes and related topics*, Lausanne, Switzerland, sept. 2008, p. 90–95, URL : http://hal.univ-brest.fr/hal-00487334.

[HDB20] Mohand HAMADOUCHE, Catherine DEZAN et Kalinka R. L. J. C. BANCO, « Reward Tuning for self-adaptive Policy in MDP based Distributed Decision-Making to ensure a Safe Mission Planning », *in* : *6th International Workshop on Safety and Security Intelligent Vehicle (SSIV)*, 2020.

[Hir+18c] Chabha HIRECHE et al., « Context/Resource-Aware Mission Planning Based on BNs and Concurrent MDPs for Autonomous UAVs », *in* : *Sensors* 18.*12* (2018), p. 4266.

[Hir+18e] Chabha HIRECHE et al., *Planification de Mission de Drone : Implémentation Logicielle/Matérielle (GDR SoC2)*, GDR SoC2, Poster, juin 2018, URL : http://hal.univ-brest.fr/hal-01844331.

[Hsa+21] Mohammed-Amine HSAINI et al., *Classifieur embarqué pour la détection d'intrusions dans le contexte des véhicules autonomes*, COMPAS, Poster, juil. 2021, URL : https://hal.archives-ouvertes.fr/hal-03373023.

[Kha+18a] Arwa KHANNOUSSI et al., « Incremental Learning of Simple Ranking Method Using Reference Profiles Models », *in* : *DA2PL'2018 : from Multiple Criteria Decision Aid to Preference Learning*, Poznan, Poland, nov. 2018, URL : https://hal.archives-ouvertes.fr/hal-01947860.

[Kha+18b] Arwa KHANNOUSSI et al., « Incremental preference elicitation for SRMP models : Application for autonomous drones », *in* : *88th Meeting of the EURO Working Group Multiple Criteria Decision Aiding*, Lisbonne, Portugal, sept. 2018, URL : https://hal-imt-atlantique.archives-ouvertes.fr/hal-01886972.

[Kha+18c] Arwa KHANNOUSSI et al., « Traceable decisions for autonomous unmanned aerial vehicles », *in* : *ROADEF 2018*, Lorient, France, fév. 2018, URL : http://hal.univ-brest.fr/hal-01740032.

[Kha+19] Arwa KHANNOUSSI et al., « Integrating Operators' Preferences into Decisions of Unmanned Aerial Vehicles : Multi-layer Decision Engine and Incremental Preference Elicitation », *in* : *6th International Conference, ADT 2019*, sous la dir. de Kristen Brent Venable SAŠA PEKEC, Algorithmic Decision Theory, Durham, NC, United States, oct. 2019, p. 49–63, DOI : 10.1007/978-3-030-31489-7_4, URL : https://hal-imt-atlantique.archives-ouvertes.fr/hal-02334090.

[Lag+02] Loic LAGADEC et al., « A LUT based Approach for High Level Synthesis on FPGAs », *in* : *International Workshop on Logic and Synthesis (IWLS)*, New Orleans, United States, juin 2002, URL : http://hal.univ-brest.fr/hal-01862801.

[Le +20] Tanguy LE PENNEC et al., « Underwater exploration by AUV using deep neural network implemented on FPGA », *in* : *Pattern Recognition and Tracking XXXI*, sous la dir. de Mohammad S. ALAM, t. 11400, International Society for Optics et Photonics, SPIE, 2020, p. 61–66, DOI : 10.1117/12.2558606, URL : https://doi.org/10.1117/12.2558606.

[Mej+21] Luis MEJIAS et al., « Embedded Computation Architectures for Autonomy in Unmanned Aircraft Systems (UAS) », *in* : *Sensors* 21.*4* (2021), ISSN : 1424-8220, DOI : 10.3390/s21041115, URL : https://www.mdpi.com/1424-8220/21/4/1115.

[Mor+07] Csaba Andras MORITZ et al., « Fault-Tolerant Nanoscale Processors on Semiconductor Nanowire Grids », *in* : *IEEE Transactions on Circuits and Systems* 54.*11* (nov. 2007), p. 2422–2437, URL : https://hal.archives-ouvertes.fr/hal-00169889.

[PR96] JL. PHILIPPE P. ASAR (NOM COLLECTIF DE : P. AUBRY ; M. BELHADJ ; TH. GAUTIE ; P. LE GUERNIC ; P. QUINTON ; C. DEZAN ; M. ISRAEL ; J. BENZAKKI ; T. BOUGUERBA ; F. ROUSSEAU ; M. AUGUIN ; C. CARRIERE ; G. COGNIAT ; G. DURRIEU ; M. LEMAITRE ; E. MARTIN ; O. SENTIEYS et L. RIDEAU),

« Vers un Atelier d'accueil générique pour la Synthése ARchitecturale bâti autour de Centaur : ASAR », *in* : *Quatrième Symposium Architectures Nouvelles de Machines*, IRISA, 1996, p. 51–62.

[Rib+18] Vincent Ribaud et al., « Eduquer en montrant l'exemple : les filles qui ... », *in* : *Didapro 7 – DidaSTIC : De 0 à 1 ou l'heure de l'informatique à l'école*, Lausanne, Switzerland, fév. 2018, URL : http://hal.univ-brest.fr/hal-01756177.

[San+18] Guilherme Santana et al., « Cognitive Radio for UAV communications :Opportunities and future challenges », *in* : *International Conference on Unmanned Aircraft Systems (ICUAS'18)*, Dallas, United States, juin 2018, URL : http://hal.univ-brest.fr/hal-01842441.

[San+19] Guilherme Santana et al., « A Case Study of Primary User Arrival Prediction Using the Energy Detector and the Hidden Markov Model in Cognitive Radio Networks », *in* : *WoCCES 2019*, Barcelone, Spain, juin 2019, URL : https://hal.univ-brest.fr/hal-02389059.

[TDL08] Ciprian Teodorov, Catherine Dezan et Loïc Lagadec, « On the Way to Design Computing Architectures with Emerging Nanoscale Technologies », *in* : *Colloque annuel GDR-SOC-Sip*, Paris, France, juin 2008, URL : http://hal.univ-brest.fr/hal-00380878.

[Yaz+09] Samar Yazdani et al., « Optimizing Memory Access Latencies on a Reconfigurable Multimedia Accelerator : A Case of a Turbo Product Codes Decoder », *in* : *ARC 2009, the 5th International Workshop on Applied Reconfigurable Computing*, t. 5453, Lecture Notes in Computer Science, Germany : Springer Berlin / Heidelberg, mar. 2009, p. 287–292, DOI : 10.1007/978-3-642-00641-8_30, URL : http://hal.univ-brest.fr/hal-00490480.

[ZD19] Sara Zermani et Catherine Dezan, « Generation of a Reconfigurable Probabilistic Decision-Making Engine based on Decision Networks : UAV Case Study (Interactive Presentation) », *in* : *Workshop on Autonomous Systems Design, ASD 2019, March 29, 2019, Florence, Italy*, 2019, 9 :1–9 :14, DOI : 10.4230/OASIcs.ASD.2019.9, URL : https://doi.org/10.4230/OASIcs.ASD.2019.9.

[ZDE17b] Sara Zermani, Catherine Dezan et Reinhardt Euler, « Embedded Decision Making for UAV Missions », *in* : *6th Mediterranean Conference on Embedded Computing (MECO)*, Bar, Montenegro, juin 2017, URL : https://hal.archives-ouvertes.fr/hal-01528309.

[Zer+16d] Sara Zermani et al., *SWARMS Project : Self-Adaptive HW/SW Architecture for Unmanned Aerial Vehicles (UAVs)(Séminaire des doctorantes et doctorants de la SIF, GDR SoC-SiP)*, Séminaire des doctorantes et doctorants de la SIF, GDR SoC-SiP, Poster, avr. 2016, URL : http://hal.univ-brest.fr/hal-01844370.

Autres références scientifiques

[All+19] Azza Allouch et al., « Qualitative and Quantitative Risk Analysis and Safety Assessment of Unmanned Aerial Vehicles Missions Over the Internet », *in* : *IEEE Access* 7 (2019), p. 53392–53410, DOI : 10.1109/ACCESS.2019.2911980.

[Cai+18] Baoping Cai et al., « Application of Bayesian networks in reliability evaluation », *in* : *IEEE Transactions on Industrial Informatics* 15.*4* (2018), p. 2146–2157.

[Cir11] ICAO Cir, « 328 AN/190 », *in* : *Unmanned Aircraft Systems (UAS) Circular* 10 (2011), URL : https://www.icao.int/Meetings/UAS/Documents/Circular%20328_en.pdf.

[CLN13] Junyi Chai, James NK Liu et Eric WT Ngai, « Application of decision-making techniques in supplier selection : A systematic review of literature », *in* : *Expert Systems with Applications* 40.*10* (2013), p. 3872–3885.

[Clo02] Bruce T Clough, *Metrics, schmetrics ! How the heck do you determine a UAV's autonomy anyway*, rapp. tech., Air Force Research Lab Wright-Patterson AFB OH, 2002.

[Dar00] Adnan Darwiche, « A Differential Approach to Inference in Bayesian Networks », *in* : *UAI*, 2000, p. 123–132.

[Dar03] A. Darwiche, « A differential approach to inference in Bayesian networks », *in* : *J. ACM* 50.*3* (2003), p. 280–305.

[Dar09] Professor Adnan Darwiche, *Modeling and reasoning with Bayesian networks*, 1st, New York, NY, USA : Cambridge University Press, 2009, ISBN : 0521884381, 9780521884389.

[DD07] Jérémie Detrey et Florent De Dinechin, « A tool for unbiased comparison between logarithmic and floating-point arithmetic », *in* : *The Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology* 49.*1* (2007), p. 161–175.

[Dev] Yole Developpement, Accessed : 2021-12-06, URL : https://1.bp.blogspot.com/-EqHDWJT3G88/X2OvK14CfrI/AAAAAAAAgns/18TZfBtyYS09j6zGxo4DKBmSPuc
s2048/Yole-5.JPG.

[DMC19] Rocio Diaz de León Torres, Martın Molina González et Pascual Campoy Cervera, « Survey of Bayesian Network applications to Intelligent Autonomous Vehicles (IAVs) », *in* : *eprint arXiv : 1901.05517* (2019), p. 1–34.

[DSA11] Rónán Daly, Qiang Shen et Stuart Aitken, « Learning Bayesian networks : approaches and issues », *in* : *The knowledge engineering review* 26.*2* (2011), p. 99–157.

[Fad95] David S Fadok, *John Boyd and John Warden : Air Power's Quest for Strategic Paralysis*. Rapp. tech., AIR UNIV MAXWELL AFB AL SCHOOL OF ADVANCED AIRPOWER STUDIES, 1995.

[Fel20] Leonardo Carneiro Feltran, *LeNet : FPGA implementation*, Rapport de stage master, UBO, juin 2020.

[FW18] Jorge F Figueroa et Mark G Walker, « Integrated system health management (ISHM) and autonomy », *in* : *2018 AIAA Information Systems-AIAA Infotech@ Aerospace*, 2018, p. 1152.

[GCD15] Zhiwei Gao, Carlo Cecati et Steven X Ding, « A survey of fault diagnosis and fault-tolerant techniques—Part I : Fault diagnosis with model-based and signal-based approaches », *in* : *IEEE transactions on industrial electronics* 62.*6* (2015), p. 3757–3767.

[GMG16] Philipp Gysel, Mohammad Motamedi et Soheil Ghiasi, « Hardware-oriented approximation of convolutional neural networks. CoRR abs/1604.03168 (2016) », *in* : *arXiv preprint arXiv :1604.03168* (2016).

[Hen17] Jacques Henno, *1956 : et l'intelligence artificielle devint une science*, Accessed : 2020-07-09, 2017, URL : https://www.lesechos.fr/2017/08/1956-et-lintelligence-artificielle-devint-une-science-181042.

[HK16] Rotem Ben Hur et Shahar Kvatinsky, « Memory Processing Unit for in-memory processing », *in* : *2016 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*, 2016, p. 171–172, DOI : 10.1145/2950067.2950086.

[Hua+05] Hui-Min Huang et al., « A framework for autonomy levels for unmanned systems (ALFUS) », *in* : *Proceedings of the AUVSI's unmanned systems North America* (2005), p. 849–863.

[Ken12] Farid Kendoul, « Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems », *in* : *Journal of Field Robotics* 29.*2* (2012), p. 315–378.

[Kha19] Arwa Khannoussi, « Intégration des préférences d'un opérateur dans les décisions d'un drone autonome et élicitation incrémentale de ces préférences. (Integration of an operator's preferences into the decisions of an unmanned aerial vehicle and incremental elicitation of these preferences) », thèse de doct., University of Western Brittany, Brest, France, 2019, URL : https://tel.archives-ouvertes.fr/tel-02536674.

[KKA11] Nima Khakzad, Faisal Khan et Paul Amyotte, « Safety analysis in process facilities : Comparison of fault tree and Bayesian network approaches », *in* : *Reliability Engineering & System Safety* 96.*8* (2011), p. 925–932.

[KKA12] Nima Khakzad, Faisal Khan et Paul Amyotte, « Dynamic risk analysis using bow-tie approach », *in* : *Reliability Engineering & System Safety* 104 (2012), p. 36–44.

[KRM93] Ralph L Keeney, Howard Raiffa et Richard F Meyer, *Decisions with multiple objectives : preferences and value trade-offs*, Cambridge university press, 1993.

[KZZ19] Wang Kang, He Zhang et Weisheng Zhao, « Spintronic Memories : From Memory to Computing-in-Memory », *in* : *2019 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*, 2019, p. 1–2, doi : 10.1109/NANOARCH47378.2019.181298.

[Ler06] Philippe Leray, « Réseaux bayésiens : apprentissage et modélisation de systèmes complexes », *in* : *habilitation à diriger les recherches, Université de Rouen* (2006).

[Li+20] Li Li et al., « A review of applications in federated learning », *in* : *Computers & Industrial Engineering* 149 (2020), p. 106854.

[Liu+20] Jiayi Liu et al., « Pruning Algorithms to Accelerate Convolutional Neural Networks for Edge Applications : A Survey », *in* : *CoRR* abs/2005.04275 (2020), arXiv : 2005.04275, url : https://arxiv.org/abs/2005.04275.

[MAM19] Salama A Mostafa, Mohd Sharifuddin Ahmad et Aida Mustapha, « Adjustable autonomy : a systematic literature review », *in* : *Artificial Intelligence Review* 51.*2* (2019), p. 149–186.

[Maz21] Julien Mazuet, « Reconfiguration des ressources matérielles et logicielles d'un sys- tème Radar embarqué en mission d'interception », thèse de doct., Université de Bretagne Occidentale (UBO), Brest, 2021.

[MDU08] O. J. Mengshoel, A. Darwiche et S. Uckun, « Sensor Validation using Bayesian Networks », *in* : *Proceedings of the 9th International Symposium on Artificial Intelligence, Robotics, and Automation in Space (iSAIRAS-08)*, Los Angeles, CA, fév. 2008.

[min19] Etudes ministérielles, *Intelligence artificielle – État de l'art et perspectives pour la France*, Accessed : 2020-07-09, 2019, url : archive.wikiwix.com/cache/index2.php?rev_t=1594307623&url=https://cget.gouv.fr/ressources/publications/intelligence-artificielle-etat-de-l-art-et-perspectives-pour-la-france.

[Mot+16] Mohammad Motamedi et al., « Design space exploration of FPGA-based deep convolutional neural networks », *in* : *2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC)*, IEEE, 2016, p. 575–580.

[Mur02] K.P. Murphy, « Dynamic Bayesian Networks : Representation, Inference and Learning », AAI3082340, thèse de doct., 2002.

[Naï+07] P. Naïm et al., *Réseaux bayésiens*, Algorithmes, Eyrolles, 2007, p. 424, url : https://hal.archives-ouvertes.fr/hal-00412267.

[Pea88] Judea Pearl, *Probabilistic reasoning in intelligent systems : Networks of plausible inference*, San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1988, isbn : 0-934613-73-7.

[PSW00] Raja Parasuraman, Thomas B Sheridan et Christopher D Wickens, « A model for types and levels of human interaction with automation », *in* : *IEEE Transactions on systems, man, and cybernetics-Part A : Systems and Humans* 30.*3* (2000), p. 286–297.

[Ran+22] Kavindu Ranasinghe et al., « Advances in Integrated System Health Management for mission-essential and safety-critical aerospace applications », *in* : *Progress in Aerospace Sciences* 128 (2022), p. 100758.

[Reu+21] Albert Reuther et al., *AI Accelerator Survey and Trends*, 2021, arXiv : 2109.08957 [cs.AR].

[RM09] B. W. Ricks et O. J. Mengshoel, « The Diagnostic Challenge Competition : Probabilistic Techniques for Fault Diagnosis in Electrical Power Systems », *in* : *Proceedings of the 20th International Workshop on Principles of Diagnosis (DX-09)*, Stockholm, Sweden, 2009.

[Roy68] Bernard Roy, « Classement et choix en présence de points de vue multiples », *in* : *Revue française d'informatique et de recherche opérationnelle* 2.*8* (1968), p. 57–75.

[Sch+13a] Johann Schumann et al., « Towards Real-time, On-board, Hardware-supported Sensor and Software Health Management for Unmanned Aerial Systems », *in* : *Proceedings of the 2013 Annual Conference of the Prognostics and Health Management Society (PHM2013)*, oct. 2013.

[Sch+13b] J. Schumann et al., « Software health management with Bayesian networks », *in* : *Innovations in Systems and Software Engineering* 9.*4* (2013), p. 271–292.

[Sch+15] Johann M Schumann et al., « Towards real-time, on-board, hardware-supported sensor and software health management for unmanned aerial systems », *in* : *International Journal of Prognostics and Health Management* 6 (2015).

[Sho07] Eric Sholes, « Evolution of a UAV autonomy classification taxonomy », *in* : *2007 IEEE Aerospace Conference*, IEEE, 2007, p. 1–16.

[SMR15] Johann Schumann, Patrick Moosbrugger et Kristin Y Rozier, « R2U2 : Monitoring and Diagnosis of Security Threats for Unmanned Aerial Systems », *in* : *Runtime Verification*, Springer, 2015, p. 233–249.

[St 06] Tom St Denis, *BigNum Math : implementing cryptographic multiple precision arithmetic*, Elsevier, 2006.

[Str11] JR Struharik, « Implementing decision trees in hardware », *in* : *2011 IEEE 9th International Symposium on Intelligent Systems and Informatics*, IEEE, 2011, p. 41–46.

[SV78] Thomas B Sheridan et William L Verplank, *Human and computer control of undersea teleoperators*, rapp. tech., Massachusetts Inst of Tech Cambridge Man-Machine Systems Lab, 1978.

[Teo11] Ciprian Teodorov, « Model-Driven Physical-Design for Future Nanoscale Architectures », thèse de doct., Université de Bretagne Occidentale (UBO), Brest, 2011.

[TG17] Massimo Tipaldi et Luigi Glielmo, « A survey on model-based mission planning and execution for autonomous spacecraft », *in* : *IEEE Systems Journal* 12.*4* (2017), p. 3893–3905.

[Tur51] Alan Turing, *Can digital computers think ?*, Accessed : 2020-07-09, 1951, url : `http://www.turingarchive.org/browse.php/B/5`.

[VN14] Mario Vestias et Horacio Neto, « Trends of CPU, GPU and FPGA for high-performance computing », *in* : *2014 24th International Conference on Field Programmable Logic and Applications (FPL)*, IEEE, 2014, p. 1–6.

[YWL12] Zijian Yang, Yong Wang et Jiaguo Lv, « Survey of modern fault diagnosis methods in networks », *in* : *2012 International Conference on Systems and Informatics (ICSAI2012)*, IEEE, 2012, p. 1640–1643.

[ZCD15] Gao Zhiwei, Carlo Cecati et Steven X Ding, « A survey of fault diagnosis and fault-tolerant techniques—Part II : Fault diagnosis with knowledge-based and hybrid/active approaches », *in* : (2015).

[Zer17] Sara Zermani, « Implémentation sur SoC des réseaux Bayésiens pour l'état de santé et la décision dans le cadre de missions de véhicules autonomes », thèse de doct., Université de Bretagne occidentale-Brest, 2017.

[Zha+18] Chen Zhang et al., « Caffeine : Toward uniformed representation and acceleration for deep convolutional neural networks », *in* : *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 38.*11* (2018), p. 2072–2085.

[Zha+22] Tianming Zhao et al., « A Survey of Deep Learning on Mobile Devices : Applications, Optimizations, Challenges, and Research Opportunities », *in* : *Proceedings of the IEEE* 110.*3* (2022), p. 334–354.

[Zij+21] HU Zijian et al., « Relevant experience learning : A deep reinforcement learning method for UAV autonomous motion planning in complex unknown environments », *in* : *Chinese Journal of Aeronautics* 34.*12* (2021), p. 187–204.

Titre : Contribution à la conception d'accélérateurs matériels pour systèmes autonomes intelligents

Mots clés : Accélérateurs matériels, IA embarquée, Diagnostic et Décision embarquée, Véhicules autonomes

L'intelligence artificielle (IA) est entrée dans notre quotidien. L'IA est omniprésente via les systèmes de recommandations de notre navigateur web, via nos objets connectés, notre maison connectée, via nos téléphones portables...
Dans ce document, on s'intéresse à l'IA embarquée et notamment aux accélérateurs matériels qui peuvent être mis en place dans le cadre des véhicules autonomes. Trois aspects y sont plus particulièrement traités: 1) le cadre méthodologique pour la conception des accélérateurs matériels, 2) la sûreté de fonctionnement des drones autonomes et 3) la mise en place de la décision pour la planification de mission.
Pour la conception des accélérateurs matériels, de nouveaux flots de conception sont proposés afin faciliter l'intégration de nouvelles spécifications ainsi que leur compilation vers des supports matériels de type FPGA.
Les études menées dans le cadre de la sûreté de fonctionnement ont permis de montrer comment le modèle des réseaux Bayésiens pouvait contribuer au diagnostic et à la gestion des défaillances du système, et comment il pouvait être aussi embarquable sur le drone. Enfin on s'intéresse au mécanisme de décision d'une planification de mission en y intégrant ces éléments de diagnostic pour assurer une autonomie complète du drone.
Ces travaux ont permis de dresser un bilan et quelques perspectives sur les méthodes et méthodologies de conception d'accélérateurs matériels envisagées pour la gestion de missions de drones autonomes.

Title : Contribution to the design of hardware accelerators for intelligent autonomous systems

Keywords : Hardware Accelerators, Embedded AI, Diagnostics and On-Board Decision, Autonomous Vehicles

Abstract : Artificial intelligence (AI) has entered our daily lives. AI is omnipresent via the recommendation systems of our web browser, via our connected objects, our connected home, via our mobile phones...
In this document, we are interested in embedded AI and in particular in hardware accelerators that can be set up as part of autonomous vehicles. Three aspects are more particularly addressed: 1) the methodological framework for the design of hardware accelerators, 2) the operational safety of autonomous drones and 3) the implementation of the decision for mission planning.
For the design of hardware accelerators, new design streams are proposed to facilitate the integration of new specifications as well as their compilation to FPGA-type hardware media.
Studies conducted as part of operational safety have shown how the Bayesian network model can contribute to the diagnosis and management of system failures, and how it can also be embedded on the drone. Finally, we deal with the decision-making mechanism of a mission planning by integrating these diagnostic elements to ensure complete autonomy of the drone. This work made it possible to draw up an assessment and some perspectives on the methods and methodologies of design of hardware accelerators envisaged for the management of autonomous drone missions.