



Multidisciplinary collaborative design of complex systems

Moncef Hammadi

► To cite this version:

Moncef Hammadi. Multidisciplinary collaborative design of complex systems. Mechanical engineering [physics.class-ph]. Université de Technologie de Compiègne, 2021. ⟨tel-03614326⟩

HAL Id: tel-03614326

<https://hal.science/tel-03614326v1>

Submitted on 20 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Université de Technologie de Compiègne - UTC

École Doctorale

Sciences de l'Ingénieur

HDR

pour obtenir le titre de

Habilitation à diriger des recherches

de Université de Technologie de Compiègne

Spécialité : GÉNIE INDUSTRIEL

Présentée par

Moncef HAMMADI

Maître de Conférences

Docteur de l'Ecole Centrale de Paris

**Conception multidisciplinaire
collaborative de systèmes
complexes**

Préparé au laboratoire Quartz EA7393, Equipe IS2M

Soutenue publiquement le 30 novembre 2021 devant le jury composé de :

Jury :

<i>Rapporteurs :</i>	Pr. Christine PRELLE	- UTC Compiègne
	Pr. Vincent CHEUTET	- INSA-Lyon
	Pr. Abdelkhalak EL HAMI	- INSA-Rouen
<i>Examineurs :</i>	Pr. Samir LAMOURI	- ENSAM
	Pr. Frédéric KRATZ	- INSA Centre Val de Loire
	Ing. de rech. HDR. Olivia PENAS	- ISAE-Supméca
	Pr. Omar HAMMAMI	- ENSTA et Inst. Polytech. de Paris
<i>Tuteur :</i>	Pr. Jean-Yves CHOLEY	- ISAE-Supméca

Remerciements

Je tiens d'abord à remercier tout particulièrement Monsieur Jean-Yves Choley, Professeur à ISAE-Supméca, Directeur de recherche, responsable de l'équipe IS2M au laboratoire Quartz EA7393, qui, lors de ma thèse puis durant toutes ces années passées, m'a toujours accompagné dans l'évolution de ma thématique de recherche. Qu'il trouve ici ma profonde reconnaissance.

Je remercie Madame Christine Prelle, Professeur à l'UTC Compiègne, Monsieur Vincent Cheutet, Professeur à l'INSA de Lyon et Monsieur Abdelkhalak El Hami, Professeur à l'INSA de Rouen pour l'honneur qu'il m'ont fait d'être rapporteurs de ce mémoire d'habilitation.

Je tiens à remercier également Monsieur Samir Lamouri, Professeur à l'ENSAM, Monsieur Frédérique Kratz, Professeur à l'INSA Centre Val de Loire, Madame Olivia Penas, Ingénieur de recherche HDR à ISAE-Supméca et Monsieur Omar Hammami, Professeur à l'ENSTA-ParisTech qui m'ont fait l'honneur de participer au jury et d'avoir accepté d'examiner ce travail.

Enfin, je souhaite remercier tous mes collègues du laboratoire Quartz, et particulièrement l'équipe IS2M qui, par leurs nombreuses discussions et leur amitié, nous permettent de travailler dans une ambiance conviviale.

Préambule

Le présent document décrit mes activités universitaires d'enseignant-chercheur depuis ma soutenance de thèse en 2012 effectuée au Laboratoire d'Ingénierie des Systèmes Mécaniques et des Matériaux (LISMMA) à ISAE-Supméca.

Les thèmes développés en recherche concernent la formalisation des cadres de modélisation et de conception afin de maîtriser la complexité croissante des produits et des processus de production en raison d'une importante intégration multidisciplinaire au cœur même des produits et des processus de production. Plus précisément, notre travail porte sur le développement d'approches méthodologiques, fondées sur une intégration Mécatronique afin de garantir a priori le respect des différentes exigences de conception.

Notre projet de recherche propose une action centrale visant à définir un cadre formel pour un processus de conception dans un contexte d'ingénierie système, permettant une conception collaborative multidisciplinaire appliquée aux systèmes complexes de types Mécatronique, CPS et CPPS.

Ce manuscrit contient la partie scientifique du rapport HDR. Dans la première partie, je présente une synthèse de nos activités de recherche et de nos contributions. La deuxième partie détaille notre projet de recherche et les axes à développer.

Enfin, cette partie scientifique a été rédigée en anglais.

Contents

List of acronyms	v
0.1 Scientific publications	1
0.1.1 Articles in peer-reviewed journals	1
.	1
0.1.2 International communications with proceedings	5
.	5
1 Collaborative multidisciplinary design of mechatronic systems	13
1.1 Context, scientific background, positioning and research issues	14
1.1.1 Context	14
1.1.2 Scientific background	17
1.1.3 Positioning	36
1.1.4 Research issues	40
1.2 MBSE-based approaches for the integrated design of mechatronic systems	40
1.2.1 Production process modeling for mechatronic product design	40
1.2.2 Model-Based Systems Engineering (MBSE) approaches to integrate Design for X (DFX) for mechatronics design	58
1.3 Multiphysics simulation methods for mechatronics design	71
1.3.1 Contribution to the development of compact analytical models for vibration analysis of mechatronic systems	71
1.4 Interdisciplinary collaborative design of mechatronic systems	79
1.4.1 Collaborative Design Process and Product Knowledge (CDPPK) Methodology	79
1.4.2 Collaborative design assisted by multi-agent systems	89
1.5 Conclusions and synthesis	96
References	99
2 Research project and perspectives: Collaborative multidisciplinary design of Cyber Physical Systems	107
2.1 Introduction	107
2.2 New industrial challenges	107
2.3 New research challenges	108
2.4 New research axis	110
2.4.1 Research axis 1: Digital twin development for cyber physical systems	110
2.4.2 Research axis 2: Interdisciplinary collaborative design of cyber physical (production) systems	113
2.5 Conclusion	114
References	116

List of acronyms

ADACOR	Adaptive holonic architecture	96
AI	Artificial Intelligence	108
ASP	Assembly Sequence Planning	26
CAD	Computer Aided Design	33
CAE	Computer Aided Engineering	33
CDP	Collaborative Design Process	82
CDPPK	Collaborative Design Process and Product Knowledge	82
CE	Concurrent Engineering	33
CM	Configuration Management	33
CPS	Cyber Physical System	109
CPPS	Cyber Physical Production System	109
DAE	Differential Algebraic Equations	22
DFA	Design for Assembly	24
DFD	Design for Disassembly	27
DFM	Design for Manufacture	24
DFMA	Design for Manufacture and Assembly	24
DFX	Design for X	iii
DPK	Design Product Knowledge	81
DOF	Degree Of Freedom	26
DSL	Domain Specific Language	51
DSP	Disassembly Sequence Planning	27
ETB	Electronic Throttle Body	25
FDM	Finite Difference Method	24
FEM	Finite Element Method	24
FMEA	Failure Modes and Effects Analysis	62

FSI Fluid Structure Interaction	22
FVM Finite Volume Method	24
ICDKM Interdisciplinary Collaborative Design and Knowledge Management	18
ICE Information Core Entity	34
INCOSE International Council on Systems Engineering	19
IS2M Ingénierie des Systèmes Mécatroniques et Multiphysiques	41
IoT Inernet of Things	108
JADE Java Agent Development	90
KC Knowledge Configuration	34
KCM Knowledge Configuration Management	33
KM Knowledge Management	82
KVA Key Value Attributes	58
MAS Multi-Agent System	90
MBSE Model-Based Systems Engineering	iii
MDAO Multidisciplinary Design Analysis and Optimization	16
MDO Multidisciplinary Design Optimization	28
MPDT Multiphysics Digital Twin	110
MPS Multiphysics Simulation	17
MSD Mechatronic System Design	17
PD Project Domain	82
PDE Partial Differential Equations	22
PLM Product Lifecycle Management	33
ODE Ordinary Differential Equations	22
OMG Object Management Group	19
RMS Reconfigurable Manufacturing System	48
ROS Robot Operating System	113
SBCE Set-Based Concurrent Engineering	42

SysML Systems Modeling Language	16
UC Usage Configuration	34
UML Unified Modeling Language	16
UPC Usage Process Configuration	81

Scientific publications

0.1 Scientific publications

0.1.1 Articles in peer-reviewed journals

- [1] **M. Hammadi**, J.-Y. Choley, O. Penas, J. Louati, A. Rivière, and M. Haddar, “Layout optimization of power modules using a sequentially coupled approach,” *International Journal of Simulation Modelling*, vol. 10, no. 3, pp. 122–132, 2012. (Not cited.)
- [2] **M. Hammadi**, J.-Y. Choley, and F. Mhenni, “A multi-agent methodology for multi-level modeling of mechatronic systems,” *Advanced Engineering Informatics*, vol. 28, no. 3, pp. 208–217, 2014. (Not cited.)
- [3] F. Mhenni, J.-Y. Choley, O. Penas, R. Plateaux, and **M. Hammadi**, “A sysml-based methodology for mechatronic systems architectural design,” *Advanced Engineering Informatics*, vol. 28, no. 3, pp. 218–231, 2014. (Cited on page 19.)
- [4] G. Hamza, J.-Y. Choley, **M. Hammadi**, A. Riviere, M. Barkallah, J. Louati, and M. Haddar, “Pre-designing of a mechatronic system using an analytical approach with dymola,” *Journal of Theoretical and Applied Mechanics*, 2015. (Not cited.)
- [5] **M. Hammadi** and J.-Y. Choley, “Parametric compact modelling of dynamical systems using meshfree method with multi-port technique,” *International Journal of Dynamical Systems and Differential Equations*, vol. 5, no. 3, pp. 206–219, 2015. (Not cited.)
- [6] **M. Hammadi**, A. Guizani, J.-Y. Choley, A. Kellner, and P. Hehenberger, “An agent-supported approach for the collaborative design of complex systems,” *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 7, no. 02, p. 1642001, 2016. (Not cited.)
- [7] A. Guizani, **M. Hammadi**, J.-Y. Choley, T. Soriano, M. S. Abbes, and M. Haddar., “Electric vehicle design, modelling and optimization,” *Mechanics & Industry*, vol. 17, no. 4, p. 405, 2016. (Not cited.)
- [8] C. Capasso, **M. Hammadi**, S. Patalano, R. Renaud, and O. Veneri, “A multi-domain modelling and verification procedure within mbse approach to design propulsion systems for road electric vehicles,” *Mechanics & Industry*, vol. 18, no. 1, p. 107, 2017. (Not cited.)

- [9] O. Penas, R. Plateaux, S. Patalano, and **M. Hammadi**, “Multi-scale approach from mechatronic to cyber-physical systems for the design of manufacturing systems,” *Computers in Industry*, vol. 86, pp. 52–69, 2017. (Not cited.)
- [10] *G. Hamza*, **M. Hammadi**, M. Barkallah, J.-Y. Choley, A. Riviere, J. Louati, and M. Haddar, “Conceptual design methodology for the preliminary study of a mechatronic system: application to wind turbine system,” *Mechanics & Industry*, vol. 18, no. 4, p. 413, 2017. (Not cited.)
- [11] *A. Guizani*, **M. Hammadi**, J.-Y. Choley, T. Soriano, *et al.*, “Multi-agent approach based on a design process for the optimization of mechatronic systems,” *Mechanics & Industry*, vol. 18, no. 5, p. 507, 2017. (Not cited.)
- [12] R. B. Hadj, I. Belhadj, C. Gouta, M. Trigui, N. Aifaoui, and **M. Hammadi**, “An interoperability process between cad system and cae applications based on cad data,” *International journal on interactive design and manufacturing (IJIDeM)*, vol. 12, no. 3, pp. 1039–1058, 2018. (Cited on page 27.)
- [13] *G. Hamza*, **M. Hammadi**, M. Barkallah, J.-Y. Choley, A. Riviere, J. Louati, and M. Haddar, “Compact analytical models for vibration analysis in modelica/dymola: application to the wind turbine drive train system,” *Chinese Journal of Mechanical Engineering*, vol. 39, no. 1, pp. 121–130, 2018. (Not cited.)
- [14] *G. Hamza*, **M. Hammadi**, M. Barkallah, J.-Y. Choley, A. Riviere, J. Louati, and M. Haddar, “Conceptual design decision support of a mechatronic system using analytical approach with modelica,” *Mechanics & Industry*, vol. 19, no. 1, p. 103, 2018. (Not cited.)
- [15] *M. Mcharek*, T. Azib, **M. Hammadi**, J.-Y. Choley, and C. Larouci, “Knowledge sharing for mechatronic systems design and optimization,” *IFAC-PapersOnLine*, vol. 51, no. 11, pp. 1365–1370, 2018. (Not cited.)
- [16] *A. Abid*, **M. Hammadi**, M. Barkallah, J.-Y. Choley, J. Louati, A. Rivière, and M. Haddar, “Generic framework for holonic modelling and multi-agent based verification of reconfigurable manufacturing systems,” *International Journal of Precision Engineering and Manufacturing*, vol. 19, no. 12, pp. 1793–1809, 2018. (Not cited.)
- [17] *M-F. Borchani*, **M. Hammadi**, N. Ben Yahia, and J.-Y. Choley, “Integrating model-based system engineering with set-based concurrent engineering principles for reliability and manufacturability analysis of mechatronic products,” *Concurrent Engineering*, vol. 27, no. 1, pp. 80–94, 2019. (Not cited.)
- [18] *M. Mcharek*, **M. Hammadi**, T. Azib, C. Larouci, and J.-Y. Choley, “Collaborative design process and product knowledge methodology for mechatronic systems,” *Computers in Industry*, vol. 105, pp. 213–228, 2019. (Not cited.)

-
- [19] A. Bedeoui, R. B. Hadj, **M. Hammadi**, M. Trigui, and N. Aifaoui, "Assembly sequence plan generation of heavy machines based on the stability criterion," *The International Journal of Advanced Manufacturing Technology*, vol. 102, no. 9, pp. 2745–2755, 2019. (Cited on page 27.)
 - [20] R. Ammar, **M. Hammadi**, J.-Y. Choley, M. Barkallah, J. Louati, and M. Haddar, "Narrowing the set of complex systems' possible design solutions derived from the set-based concurrent engineering approach," *Concurrent Engineering*, vol. 27, no. 3, pp. 233–248, 2019. (Not cited.)
 - [21] S. B. Said, R. Benhadj, **M. Hammadi**, M. Trigui, and N. Aifaoui, "Mathematic formulation for the generation of combined paths for mounting parts in assembly," *The International Journal of Advanced Manufacturing Technology*, vol. 104, no. 9, pp. 4475–4484, 2019. (Cited on page 27.)
 - [22] M. Mcharek, T. Azib, **M. Hammadi**, C. Larouci, and J.-Y. Choley, "Multiphysical design approach for automotive electronic throttle body," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 8, pp. 6752–6761, 2019. (Not cited.)
 - [23] H. Trabelsi, A. Guizani, D. Tounsi, **M. Hammadi**, M. Barkallah, and M. Haddar, "Consideration of uncertainties in the preliminary design case of an electromagnetic spindle," *Journal of Theoretical and Applied Mechanics*, vol. 57, 2019. (Not cited.)
 - [24] M. Mcharek, T. Azib, **M. Hammadi**, C. Larouci, and J.-Y. Choley, "Multidisciplinary design optimization using knowledge management applied to an electronic throttle," *COMPEL-The international journal for computation and mathematics in electrical and electronic engineering*, 2020. (Not cited.)
 - [25] G. Hamza, M. Barkallah, **M. Hammadi**, J.-Y. Choley, A. Riviere, J. Louati, and M. Haddar, "Predesign of a flexible multibody system excited by moving load using a mechatronic system approach," *Mechanics & Industry*, vol. 21, no. 6, p. 604, 2020. (Not cited.)
 - [26] I. Belhadj, **M. Hammadi**, M. Trigui, and N. Aifaoui, "Generation of disassembly plans and quality assessment based on cad data," *International Journal of Computer Integrated Manufacturing*, vol. 33, no. 12, pp. 1300–1320, 2020. (Cited on pages 27 and 28.)
 - [27] I. Belhadj, **M. Hammadi**, and N. Aifaoui, "Parallel disassembly approach with recycling rate calculation of industrial products," *The International Journal of Advanced Manufacturing Technology*, vol. 113, no. 9, pp. 2969–2984, 2021. (Cited on page 27.)
 - [28] R. Brahmi, **M. Hammadi**, N. Aifaoui, and J.-Y. Choley, "Interoperability of cad models and sysml specifications for the automated checking of design requirements," *Procedia CIRP*, vol. 100, pp. 259–264, 2021. (Not cited.)

- [29] H. Trabelsi, A. Guizani, M. Barkallah, **M. Hammadi**, A. Hadrich, and M. Haddar, “Consideration of the uncertainty in the dimensioning of a gearbox of a wind turbine,” *Journal of Theoretical and Applied Mechanics*, vol. 59, no. 1, pp. 67–79, 2021. (Not cited.)
- [30] K. Aloui, A. Guizani, **M. Hammadi**, T. Soriano, and M. Haddar, “Integrated design methodology of automated guided vehicles based on swarm robotics,” *Applied Sciences*, vol. 11, p. 6187, jul 2021. (Not cited.)
- [31] A. Bedeoui, R. B. Hadj, **M. Hammadi**, and N. Aifaoui, “Tool workspace consideration for assembly plan generation,” *Assembly Automation*, 2021. (Not cited.)

Scientific publications

0.1.2 International communications with proceedings

- [1] **M. Hammadi**, J.-Y. Choley, A. Riviere, and M. Haddar, “Integrating radial basis functions with modelica for mechatronic design,” in *Design and Modeling of Mechanical Systems*, pp. 19–25, Springer, Berlin, Heidelberg, 2013. (Not cited.)
- [2] A. Guizani, **M. Hammadi**, J.-Y. Choley, T. Soriano, M. S. Abbes, and M. Haddar, “Multidisciplinary optimization of mechatronic systems: Application to an electric vehicle,” in *Mechatronic Systems: Theory and Applications*, pp. 1–14, Springer, Cham, 2014. (Not cited.)
- [3] A. Abid, M. Barkallah, **M. Hammadi**, J.-Y. Choley, J. Louati, A. Rivière, and M. Haddar, “Conceptual design of an intelligent welding cell using sysml and holonic paradigm,” in *Design and Modeling of Mechanical Systems-II*, pp. 3–10, Springer, Cham, 2015. (Not cited.)
- [4] G. Hamza, J.-Y. Choley, **M. Hammadi**, M. Barkallah, J. Louati, A. Riviere, and M. Haddar, “Pre-dimensioning of the dynamic properties of the wind turbine system using analytical approach,” in *Design and Modeling of Mechanical Systems-II*, pp. 179–188, Springer, Cham, 2015. (Not cited.)
- [5] G. Hamza, M. Barkallah, **M. Hammadi**, J.-Y. Choley, A. Riviere, J. Louati, and M. Haddar, “An analytical approach for modeling a multibody system during pre-design with application to the railway system,” in *Advances in Materials, Mechanics and Manufacturing*, pp. 150–157, Springer, Cham, 2020. (Not cited.)
- [6] **M. Hammadi**, J.-Y. Choley, O. Penas, A. Riviere, J. Louati, and M. Haddar, “A new multi-criteria indicator for mechatronic system performance evaluation in preliminary design level,” in *Mechatronics (MECATRONICS), 2012 9th France-Japan & 7th Europe-Asia Congress on and Research and Education in Mechatronics (REM), 2012 13th Int’l Workshop on*, pp. 409–416, IEEE, 2012. (Not cited.)
- [7] **M. Hammadi**, J.-Y. Choley, O. Penas, and A. Riviere, “Mechatronic system optimization based on surrogate models-application to an electric vehicle,” in *Simultech*, pp. 11–16, 2012. (Not cited.)

- [8] **M. Hammadi**, J.-Y. Choley, O. Penas, and A. Riviere, “Multidisciplinary approach for modelling and optimization of road electric vehicles in conceptual design level,” in *2012 Electrical Systems for Aircraft, Railway and Ship Propulsion*, pp. 1–6, IEEE, 2012. (Not cited.)
- [9] **M. Hammadi**, A. Kellner, J.-Y. Choley, and P. Hehenberger, “Mechatronic design optimization using multi-agent approach,” in *The 14th Mechatronics Forum International Conference Karlstad, Sweden, June 16-18*, pp. 310–317, Leo J De Vin and Jorge Solis, 2014. (Not cited.)
- [10] **M. Hammadi**, F. Mhenni, J.-Y. Choley, A. Kellner, and P. Hehenberger, “Agent-based approach for partitioning the mechatronic design,” in *The 14th Mechatronics Forum International Conference, Karlstad-SWEDEN, June, 16-18, 2014*, pp. 622–628, Leo J De Vin and Jorge Solis, 2014. (Not cited.)
- [11] A. Guizani, **M. Hammadi**, J.-Y. Choley, T. Soriano, M. S. Abbes, and M. Haddar, “Multidisciplinary approach for optimizing mechatronic systems: Application to the optimal design of an electric vehicle,” in *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 56–61, IEEE, 2014. (Not cited.)
- [12] A. Guizani, **M. Hammadi**, J.-Y. Choley, T. Soriano, M. S. Abbes, and M. Haddar, “Agent-based approach for the optimal design of mechatronic systems,” in *Conference on Multiphysics Modelling and Simulation for Systems Design*, pp. 189–198, Springer, Cham, 2014. (Not cited.)
- [13] C. Capasso, **M. Hammadi**, S. Patalano, R. Renaud, and O. Veneri, “Rflp approach in the designing of power-trains for road electric vehicles,” in *Conference on Multiphysics Modelling and Simulation for Systems Design*, pp. 249–258, Springer, Cham, 2014. (Not cited.)
- [14] Y. B. Messaoud, **M. Hammadi**, T. Hassine, and O. Hammami, “Multidisciplinary optimization of a quad-rotor by integrating multi-level models,” in *2014 10th France-Japan/8th Europe-Asia Congress on Mechatronics (MECATRONICS2014-Tokyo)*, pp. 337–342, IEEE, 2014. (Not cited.)
- [15] A. Guizani, **M. Hammadi**, J.-Y. Choley, T. Soriano, M. S. Abbes, and M. Haddar, “Agent-based approach for collaborative distributed mechatronic design,” in *2014 10th France-Japan/8th Europe-Asia Congress on Mechatronics (MECATRONICS2014-Tokyo)*, pp. 156–161, IEEE, 2014. (Not cited.)
- [16] G. Hamza, M. Barkallah, J. Louati, M. Haddar, **M. Hammadi**, J.-Y. Choley, and A. Riviere, “Analytical approach for the integrated preliminary analysis of mechatronic systems subjected to vibration,” in *2014 10th France-Japan/8th Europe-Asia Congress on Mechatronics (MECATRONICS2014-Tokyo)*, pp. 151–155, IEEE, 2014. (Not cited.)

-
- [17] A. François, A. Lanthony, **M. Hammadi**, S. Patalano, and O. Veneri, “Placis: a general project-based learning framework illustrated with a concrete example,” in *Congrès français de mécanique*, AFM, Association Française de Mécanique, 2015. (Not cited.)
 - [18] A. François, A. Lanthony, A. Brunner, **M. Hammadi**, J.-Y. Choley, S. Patalano, and O. Veneri, “Placis: Systems engineering through a project-based learning approach general framework, debates and achievements through an overview and a concrete example,” in *2015 IEEE International Symposium on Systems Engineering (ISSE)*, pp. 358–363, IEEE, 2015. (Not cited.)
 - [19] **M. Hammadi**, M. Iebba, F. C. Rosa, O. Veneri, C. Capasso, S. Patalano, and G. Copertino, “System approach to the pre-design of electric propulsion systems for road vehicles,” in *2015 IEEE International Symposium on Systems Engineering (ISSE)*, pp. 229–236, IEEE, 2015. (Not cited.)
 - [20] R. Plateaux, O. Penas, P. Hehenberger, **M. Hammadi**, F. Mhenni, A. Warniez, and J.-Y. Choley, “Needs for a 3d enriched ontology for mechatronic systems design,” in *2015 IEEE International Symposium on Systems Engineering (ISSE)*, pp. 253–260, IEEE, 2015. (Not cited.)
 - [21] E.-M. Azzouzi, M. Iuliano, F. Camargo-Rosa, S. Patalano, **M. Hammadi**, O. Veneri, C. Capasso, and I. Motori, “Systems engineering approach for eco-comparison among power-train configurations of hybrid bus,” in *2016 Annual IEEE Systems Conference (SysCon)*, pp. 1–6, IEEE, 2016. (Not cited.)
 - [22] *M. Mcharek*, **M. Hammadi**, J.-Y. Choley, T. Azib, and C. Larouci, “Modeling and multi-objective optimization of an electronic throttle in open-loop,” in *2016 11th France-Japan & 9th Europe-Asia Congress on Mechatronics (MECATRONICS)/17th International Conference on Research and Education in Mechatronics (REM)*, pp. 331–335, IEEE, 2016. (Not cited.)
 - [23] *A. Abid*, **M. Hammadi**, J.-Y. Choley, A. Rivière, M. Barkallah, J. Louati, and M. Haddar, “A sysml based-methodology for modelling disturbances in manufacturing systems using adacor holonic control architecture,” in *2016 11th France-Japan & 9th Europe-Asia Congress on Mechatronics (MECATRONICS)/17th International Conference on Research and Education in Mechatronics (REM)*, pp. 097–102, IEEE, 2016. (Not cited.)
 - [24] R. Plateaux, O. Penas, J.-Y. Choley, F. Mhenni, **M. Hammadi**, and F. Louni, “Evolution from mechatronics to cyber physical systems: An educational point of view,” in *2016 11th France-Japan & 9th Europe-Asia Congress on Mechatronics (MECATRONICS)/17th International Conference on Research and Education in Mechatronics (REM)*, pp. 360–366, IEEE, 2016. (Not cited.)
 - [25] I. Viksne, A. Brunner, **M. Hammadi**, K. Norstrom, A. Laakkonen, W. Van der Hoeven, S. Patalano, A. Melnikovs, A. Lanthony, and A. Francois, “Epices: Assessment challenges of problem-and project-based engineering education,” in

- Proceedings of the 44th Annual Conference of the European Society for Engineering Education—Engineering Education on Top of the World: Industry-University Cooperation*, 2016. (Not cited.)
- [26] A. Abid, **M. Hammadi**, J.-Y. Choley, A. Rivière, M. Barkallah, J. Louati, and M. Haddar, “A holonic-based method for design process of cyber-physical reconfigurable systems,” in *2016 IEEE International Symposium on Systems Engineering (ISSE)*, pp. 1–5, IEEE, 2016. (Not cited.)
- [27] E.-M. Azzouzi, M. Iuliano, **M. Hammadi**, R. Plateaux, and F. Marchand, “Process integration and design optimization technologies for modelling improvement,” in *2016 IEEE International Symposium on Systems Engineering (ISSE)*, pp. 1–5, IEEE, 2016. (Not cited.)
- [28] **M. Hammadi**, J.-Y. Choley, M. A. Ben Said, A. Kellner, and P. Hehenberger, “Systems engineering analysis approach based on interoperability for reconfigurable manufacturing systems,” in *2016 IEEE International Symposium on Systems Engineering (ISSE)*, pp. 1–6, IEEE, 2016. (Not cited.)
- [29] **M. Hammadi** and S. Patalano, “Automatic generation of simulation workflows for system verification using xdsml representation,” in *Systems Engineering Symposium (ISSE), 2017 IEEE International*, 2017. (Not cited.)
- [30] M.-F. Borchani, **M. Hammadi**, N. B. Yahia, and J.-Y. Choley, “The usage of a system engineering approach for integrating machining constraints in the upstream design stage,” in *2017 IEEE International Systems Engineering Symposium (ISSE)*, pp. 1–7, IEEE, 2017. (Not cited.)
- [31] R. Ammar, **M. Hammadi**, J.-Y. Choley, M. Barkallah, J. Louat, and M. Haddar, “Architectural design of complex systems using set-based concurrent engineering,” in *2017 IEEE International Systems Engineering Symposium (ISSE)*, pp. 1–7, IEEE, 2017. (Not cited.)
- [32] R. Ammar, **M. Hammadi**, J.-Y. Choley, M. Barkallah, and J. Louati, “Mechatronic system design with manufacturing constraints using set-based concurrent engineering,” in *2018 Annual IEEE International Systems Conference (SysCon)*, pp. 1–7, IEEE, 2018. (Not cited.)
- [33] F. Mhenni, O. Penas, **M. Hammadi**, J.-Y. Choley, and P. Hehenberger, “Systems engineering approach for the conjoint design of mechatronic products and their manufacturing systems,” in *2018 Annual IEEE International Systems Conference (SysCon)*, pp. 1–8, IEEE, 2018. (Not cited.)
- [34] G. Hamza, **M. Hammadi**, M. Barkallah, J.-Y. Choley, A. Riviere, J. Louati, and M. Haddar, “Analytical based approach for vibration analysis in modelica: Application to the bridge crane system,” in *International Conference on Acoustics and Vibration*, pp. 83–91, Springer, Cham, 2018. (Not cited.)

-
- [35] *R. Ammar*, **M. Hammadi**, J.-Y. Choley, M. Barkallah, J. Louati, and M. Haddar, “The design and modeling of an optimized mechatronic system using a set based concurrent engineering,” in *International Conference on Acoustics and Vibration*, pp. 111–120, Springer, Cham, 2018. (Not cited.)
 - [36] *M-F. Borchani*, *R. Ammar*, **M. Hammadi**, J.-Y. Choley, N. B. Yahia, M. Barkallah, and J. Louati, “Mechatronic system design using model-based systems engineering and set-based concurrent engineering principles,” in *2018 12th France-Japan and 10th Europe-Asia Congress on Mechatronics*, pp. 32–38, IEEE, 2018. (Not cited.)
 - [37] *M. Mcharek*, T. Azib, **M. Hammadi**, C. Larouci, and J.-Y. Chloey, “Validation approach of a mechatronic system controller in upstream phase,” in *2018 12th France-Japan and 10th Europe-Asia Congress on Mechatronics*, pp. 212–216, IEEE, 2018. (Not cited.)
 - [38] **M. Hammadi** and O. Hammami, “System of systems architectural design of on-demand electric aviation based on mission analysis,” in *2018 IEEE International Systems Engineering Symposium (ISSE)*, pp. 1–5, IEEE, 2018. (Not cited.)
 - [39] **M. Hammadi**, M. El Idrissi, R. Plateaux, O. Penas, A. Lanthony, and I. Viksne, “Involving mechatronic students in business innovation using project-based learning: a case-study,” in *2019 20th International Conference on Research and Education in Mechatronics (REM)*, pp. 1–7, IEEE, 2019. (Not cited.)
 - [40] *R. Ammar*, **M. Hammadi**, J.-Y. Choley, M. Barkallah, and J. Louati, “Integrating the sbce principles with mbse methodology for the mechatronic system’s design,” in *2019 IEEE International Systems Conference (SysCon)*, pp. 1–6, IEEE, 2019. (Not cited.)
 - [41] *M. Mcharek*, T. Azib, C. Larouci, and **M. Hammadi**, “Collaboration and multidisciplinary design optimization for mechatronic systems,” in *IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, pp. 624–629, IEEE, 2019. (Not cited.)
 - [42] *A. Guizani*, H. Trabelsi, **M. Hammadi**, J. Choley, M. Barkallah, and M. Haddar, “Design and modeling of a mechatronic power system of an electric vehicle,” in *International Conference Design and Modeling of Mechanical Systems*, pp. 842–852, Springer, Cham, 2019. (Not cited.)
 - [43] *G. Hamza*, M. Barkallah, **M. Hammadi**, J.-Y. Choley, J. Louati, A. Riviere, and M. Haddar, “An analytical approach to model-based parametric design of mechatronic systems with modelica: A case study,” in *International Conference Mechatronics*, pp. 11–18, Springer, Cham, 2019. (Not cited.)

- [44] A. Guizani, H. Trabelsi, M. Barkallah, **M. Hammadi**, J. Choley, and M. Haddar, "Dimensioning of an electromagnetic spindle with interval computation technique," in *International Conference Mechatronics*, pp. 67–75, Springer, Cham, 2019. (Not cited.)
- [45] K. Aloui, **M. Hammadi**, T. Soriano, A. Guizani, and M. Haddar, "On the continuity of the swarm robot design using mbse method and simulation," in *13th International Conference on Modelling, Optimization and Simulation (MOSIM'20)*, 2020. (Not cited.)
- [46] R. Brahmi, **M. Hammadi**, J.-Y. Choley, M. Trigui, and N. Aifaoui, "A sysml profile for mechanical assembly," in *2020 IEEE International Systems Conference (SysCon)*, pp. 1–7, IEEE, 2020. (Not cited.)
- [47] H. Kooli-Chaabane, N. Peyret, **M. Hammadi**, and A. Lanthony, "Virtual design office: Proposition of problem-and project-based learning solution in the covid-19 era and beyond," in *2021 IEEE Global Engineering Education Conference (EDUCON)*, pp. 1527–1531, IEEE, 2021. (Not cited.)
- [48] K. Aloui, A. Guizani, **M. Hammadi**, M. Haddar, and T. Soriano, "A top down approach to ensure the continuity of the different design levels of swarm robots," in *2021 18th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 1438–1445, IEEE, 2021. (Not cited.)
- [49] R. Brahmi, **M. Hammadi**, N. Aifaoui, and J.-Y. Choley, "Cad-mbse interoperability for the checking of design requirements," in *2021 18th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 1446–1451, IEEE, 2021. (Not cited.)
- [50] A. Allagui, I. Belhadj, N. Aifaoui, **M. Hammadi**, and J.-Y. Choley, "A cad-system engineering interoperability by enriching cad database with functional information," in *2021 18th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 1452–1458, IEEE, 2021. (Not cited.)
- [51] K. Aloui, A. Guizani, **M. Hammadi**, T. Soriano, and M. Haddar, "System level specification and multi-agent simulation of manufacturing systems," in *the third edition of the international conference on Advanced Materials, Mechanics and Manufacturing*, 2021. (Not cited.)
- [52] K. Aloui, **M. Hammadi**, A. Guizani, T. Soriano, and M. Haddar, "Modeling and simulation of a swarm robot application using mbse method and multi-agent technology: Monitoring oil spills," in *The first" International Workshop on MOdelling and Simulation of COmplex Systems for Sustainable Energy Efficiency" MOSCOSSEE'2021*, 2021. (Not cited.)
- [53] K. Ueki, R. Sato, A. Ming, M. Shimojo, **M. Hammadi**, and J.-Y. Choley, "Development of knee joint mechanism with variable transmission and joint stop

for bipedal robot inspired by human structure,” in *21st International Conference on Research and Education in Mechatronics (REM)*, IEEE, dec 2020. (Not cited.)

Collaborative multidisciplinary design of mechatronic systems

Contents

1.1	Context, scientific background, positioning and research issues	14
1.1.1	Context	14
1.1.2	Scientific background	17
1.1.3	Positioning	36
1.1.4	Research issues	40
1.2	MBSE-based approaches for the integrated design of mechatronic systems	40
1.2.1	Production process modeling for mechatronic product design	40
1.2.2	MBSE approaches to integrate DFX for mechatronics design	58
1.3	Multiphysics simulation methods for mechatronics design .	71
1.3.1	Contribution to the development of compact analytical models for vibration analysis of mechatronic systems	71
1.4	Interdisciplinary collaborative design of mechatronic systems	79
1.4.1	Collaborative Design Process and Product Knowledge (CDPPK) Methodology	79
1.4.2	Collaborative design assisted by multi-agent systems	89
1.5	Conclusions and synthesis	96
	References	99

My research activities started in 2008 during my cotutelle thesis which was carried out at LA2MP/ENIS-Sfax, Tunisia and LISMMA/ISMEP-Supméca, France, then they continued without interruption in LISMMA at ISMEP-Supméca and today within Quartz EA7393. In this chapter, I present the synthesis of the research works in which I participated, in order to highlight my scientific contributions.

1.1 Context, scientific background, positioning and research issues

1.1.1 Context

The term "Mechatronics" was first introduced in 1969 by an engineer from the Japanese company Yaskawa Electric Corporation. The definition of mechatronics is given in the NF E01-010 standard (NF 2008) as *"an approach aiming at the synergistic integration of mechanics, electronics, automation and computer science in the design and manufacture of a product in order to increase and/or optimize its functionality"*.

This definition highlights the fact that functional performance of physical products is achieved through a synergistic integration of components based on heterogeneous technologies. These products are commonly called mechatronic systems, complex systems or complex engineered systems, to distinguish engineered systems from other complex systems such as natural, biological or economic systems. The mechatronics definition highlights also the fact that mechatronics is a design approach that takes into consideration the manufacturability, and more broadly speaking producibility, of the product. Finally, the definition of mechatronics emphasizes the multidisciplinary design optimization of the mechatronic system. The terms used above are the main keywords around which my research activities have been built.

The applications concerned by mechatronics touch various complex engineered systems including automobile, aerospace, naval architecture, electronics, electricity distribution, production systems, assembly/disassembly lines, just to name a few. Modern engineered systems include mechatronic modules in order to increase and optimize their functionalities. The objective is to meet the needs of increasingly demanding customers who want products cheaper, more customized, higher quality and more reliable, while at the same time being increasingly smart and innovative with high added value, while development times must be ever shorter. The design process of mechatronic systems includes activities of modeling, simulation, optimization, prototyping and deployment.

Modeling is a preliminary step that allows to create a representation of the considered system, and to appropriate its properties and specificities. Models are developed from different points of view, for different levels of abstraction, for different disciplines and physics, with different languages and tools, and by different developers from different engineering domains.

Simulation models have the particularity of allowing the analysis of the model in action, being able to inspect the behavior of the system, study its elements, analyze the processes and interact with the simulation model in action. The physical behavior of these systems is complex because it is due to the coupling of several

physics. Multiphysics is a practice built on mathematics, physics, application, and numerical analysis. The mathematics involved usually contains partial differential equations, ordinary differential equations and/or differential algebraic equations.

The optimization of a multidisciplinary system requires incorporating all relevant disciplines simultaneously, because the optimum of the simultaneous design problem is not equal to the design found by optimizing each discipline sequentially, since we can exploit the interactions between the disciplines. However, including all disciplines simultaneously significantly increases the complexity of the design problem.

Prototyping aims to make an original model that has all the technical qualities and operating characteristics of a new product. It allows to validate the choices made during the design of the units. It also allows to exchange or acquire experience on a concept or material product. As well as to test the reactions of future users or consumers and to prepare the technical specifications of the product.

The process of designing complex engineered systems requires also collaboration and knowledge management to perceive, identify, analyze, organize, memorize and share the knowledge of the members implied in the design process.

Looking at the set of design activities below, it is easy to deduce that the design process of mechatronic systems is a complex process that requires the implementation of efficient design methods and the development of new tools to manage this complexity. To achieve this, the main issue is the numerical continuity between the design stages, taking into account the evolution of the requirements throughout the design process, while facilitating collaboration between teams and engineering members.

The issue of managing the design of complex systems is not new. Classical system design cycles cover the phases of customer needs analysis, system requirements generation, functional analysis leading to functional architecture, then logical and physical architecture, as well as simultaneous pre-sizing activities allowing the evaluation of different architectures for the selection of a solution. This classical approach reveals many discontinuities when moving from one step to the next. These discontinuities are due to several factors, including the type of model needed at each level of abstraction, the level of detail of the models, the languages and tools used to develop the models, the interoperability problems between the models and the tools that support them, the difficulty of getting design members and teams with different engineering cultures to work together, etc. Thus, a disjunction is reflected in the disciplinary "Silos" in which engineers are trained, think and work.

Notable improvements over the past twenty years include the development of languages and tools for Model-Based Systems Engineering (MBSE) and

simulation. For example, the Systems Modeling Language (Systems Modeling Language ([SysML](#))) offers systems engineers several notable improvements over the more software-oriented Unified Modeling Language ([UML](#)). The semantics of [SysML](#) are richer and more flexible, imposing fewer restrictions related to the [UML](#) view, and adding new types of diagrams, such as the requirements diagram, which can be used for requirements management, while the parametric diagram can be used for performance analysis and quantitative analysis.

However, parametric diagrams only support simple mathematical models defined by algebraic equations. They do not allow, for example, the modeling of dynamic systems defined with differential equations.

Modelica is also an object-oriented modeling language for the practical modeling of complex systems. It allows a system to be described in the form of a set of equations. The associated simulator has the task of solving the system of equations at each time step. Modelica is able to describe the evolution of dynamic systems that are described by ordinary differential equations, algebraic equations and discrete equations.

The Modelica solver, on the other hand, does not support partial differential equations, which are necessary to describe the mono-physical or multi-physical behaviors of system's components. For this type of modeling, it is more appropriate to use modeling tools based on discretization methods (finite elements, finite differences and finite volumes).

Furthermore, Multidisciplinary Design Analysis and Optimization (Multidisciplinary Design Analysis and Optimization ([MDAO](#))) is a methodology that enables the analysis and optimization of a complete complex system by explicitly considering the important interactions and synergies between disciplines. [MDAO](#) helps in creating automated workflows that incorporate analysis tools and processes from different disciplines. These workflows can be run repeatedly to quickly explore and quantify the performance, costs and risks of many different design solutions. [MDAO](#) also allows sensitivity analysis to find the most important variables. In addition, [MDAO](#) workflows can be automated using optimization algorithms to find the best designs. Finally, probabilistic analyses can be performed to evaluate the robustness and reliability of the designs.

However, the implementation of [MDAO](#) workflows to automate multidisciplinary analysis and optimization is often costly. In addition, it is often necessary for decisions to be made by experts and not automatically by algorithms. This requires the implementation of effective collaborative approaches to integrate the [MDAO](#) discipline into the mechatronics design process.

Prototyping allows the designer to validate the functionality and performance of the mechatronic design, but also to verify its manufacturability, producibility, reliability, etc. Indeed, for most companies, the manufacturing system is a valuable asset that is more or less fixed and allows only minor modifications. Therefore,

the mechatronic product design must be adapted to the manufacturing system. However, trade-offs must be made when the functionality and the performance objectives of the product are affected. To make these decisions, the designer must know the existing and future manufacturing system and understand the implication of the manufacturing system on the product design. Incorporating manufacturing, production and more globally DFX constraints into the preliminary design phases of the mechatronic product will give the designer a better chance of achieving an optimal design. This will require a good specification of the mechatronic product according to the different points of view related to its functionality, performance, manufacturability, producibility, etc. It also requires effective collaboration between different multidisciplinary stakeholders, using a platform allowing efficient communication and knowledge sharing.

1.1.2 Scientific background

Our contributions in the last 10 years have focused on the development of methodologies for the design of mechatronic systems. A methodology is defined as a collection of related processes, methods and tools to solve a specific problem [Martin, 2020]. For this purpose, I have divided my research activities in this manuscript into three interrelated lines of research:

- MBSE-based approaches for the integrated design of mechatronic systems.
- Multiphysics simulation methods for mechatronics design.
- Interdisciplinary collaborative design of mechatronic systems.

Mechatronic systems here include mechatronic products and their manufacturing/production systems. The goal of the integrated design for us is to ensure continuity in the design process between the systems engineering level and the domain specific engineering activities. Domain specific engineering activities in our researches concern the Multiphysics Simulation (MPS), Multidisciplinary Design Analysis and Optimization (MDAO) and design for X (DFX) activities. The objective of the Interdisciplinary Collaborative Design and Knowledge Management (ICDKM) research activities is to provide a framework for effective collaboration between systems engineers, mechatronics engineers and specific domain specialists in multiphysics simulation, MDAO and DFX.

Figure 1.1 shows a more general view on the research fields concerned by our research activities.

In the following sections, I will start by presenting the background and the basic research issues concerning these topics and then I present our contributions. The concerned research topics are the following:

- Model-Based Systems Engineering (MBSE)
- Mechatronic System Design (MSD)

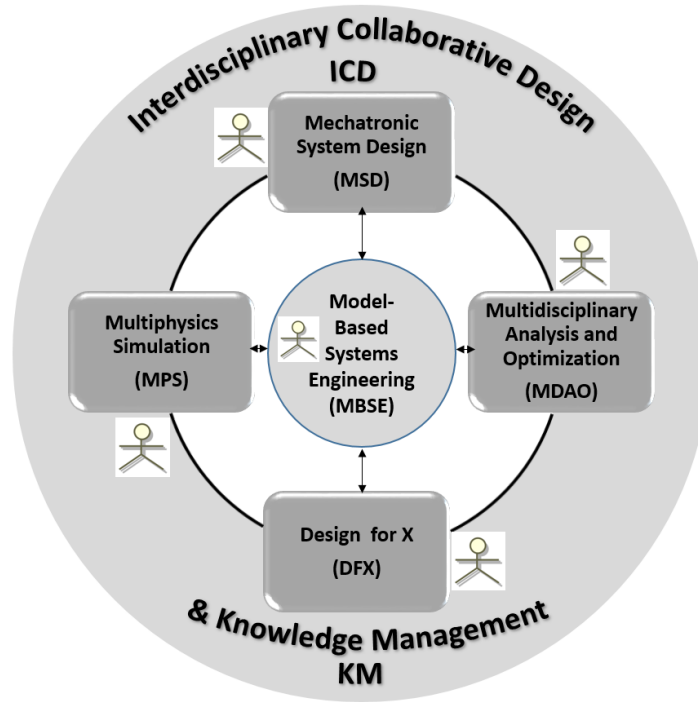


Figure 1.1: Research topics

- Multiphysics Simulation ([MPS](#))
- Multidisciplinary Design Analysis and Optimization ([MDAO](#))
- Design for X ([DFX](#))
- Interdisciplinary Collaborative Design and Knowledge Management ([ICDKM](#))

1.1.2.1 Model-Based Systems Engineering (MBSE)

With the goal of replacing paper documents with digital and data models, the use of Model-Based Systems Engineering ([MBSE](#)) has become widespread in industrial processes, and has thus improved reuse and collaboration between the different actors. Indeed, often used by system architects, it effectively encourages communication and collaboration between disciplinary experts during the different design stages. The integration of these different multidisciplinary points of view in the numerical models then contributes to the decision arbitrations. In addition, MBSE approaches increase data model consistency between disciplines, improve traceability to requirements and also promote model reuse. However, system architects need a language that allows them to specify all system requirements and architectures, regardless of the discipline or technical team they are addressing. For this purpose, [SysML](#) is generally used to implement the essential aspects of the [MBSE](#) approach. Indeed, this graphical notation language, based on [UML](#)

2.0, was initiated by the Object Management Group (OMG) and International Council on Systems Engineering (INCOSSE) in 2001, and standardized since 2006 [Hause et al., 2006]. Today, SysML is recognized by the systems engineering community as the most suitable language for implementing MBSE approaches. Mainly dedicated to systems modeling, it replaces the numerous spreadsheets used in the industry, whose consistency and updating are often difficult to maintain. Indeed, SysML allows to specify and analyze the structure and the functionality of the system and then define the system and its subsystems. Moreover, SysML allows to verify and validate the feasibility of a system before its realization by offering a unique model at the system level. In this context, each designer can refer to a single model throughout the design cycle, and has an automatic update of the data shared by all the other actors of the design. This language has the advantage of offering a great semantic richness, as it allows to represent different views of the system thanks to its different diagrams and is particularly efficient to model the requirements, structure, behavior, allocations and constraints of the system [Mhenni et al., 2014]. Three groups of diagrams make up the SysML language: behavior diagrams, requirements diagrams and structure diagrams (figure 1.2). These diagrams allow multi-view modeling with different representations of the system linked together to ensure consistency and traceability. SysML thus enables

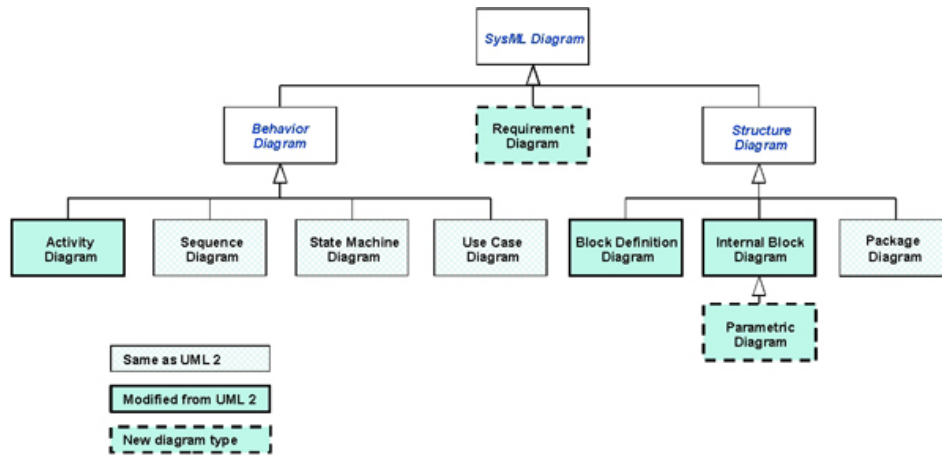


Figure 1.2: SysML Diagram Types (Source: <https://www.omgsysml.org/what-is-sysml.htm>)

the implementation of model-based systems engineering (MBSE) to allow effective communication between engineers from different disciplines.

Unfortunately, SysML is not practical enough to be used in real-world engineering projects. SysML is designed to express generic systems and requires specialized knowledge of specific engineering disciplines. Thus a model written in SysML is less able to support communication between a systems engineer and engineers from specific engineering disciplines such as Mechatronic system design, 3D design,

multiphysics simulation, MDAO and manufacturing engineers.

For this reason, new research approaches are required to overcome the weakness of the standard SysML by adapting it with the specific domain knowledge of the engineering disciplines mentioned above.

1.1.2.2 Mechatronic System Design (MSD)

The aim of Mechatronics is the synergistic integration of mechanics, electronics, automation and software in the design and manufacture of products. The synergistic integration can be achieved with a strong functional-physical system integration.

Functional integration is the simultaneous implementation of several functions within the mechatronic system. It consists in integrating as many functions as possible in a product, either by combining several functions in a single component, or by dematerializing some functions by using programmable computer technologies, which provide the system with additional sensing, communication, information processing and feedback functions to the basic mechanical functions.

Physical integration allows a set of components to be spatially integrated to form a complex functional unit. This will allow a greater reliability for a smaller installation space due to reduced interfaces and, consequently, faster data transmission. The power output will also be higher with reduced assembly effort.

The figure 1.3 shows an example of physical and functional integration concerning an electric wheel drive system where the motor, the sensors, and the power electronics are spatially integrated.

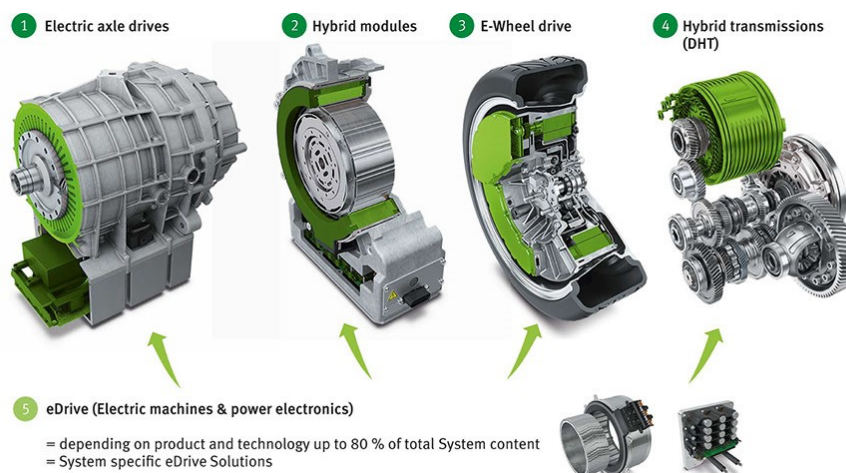


Figure 1.3: E-wheel drive: example of spatial and functional integration (Source: <https://schaeffler-events.com/>)

However, the spatial proximity of the components will generate undesirable interactions, such as heating, stray magnetic fields, vibrations, noise and voltage spikes, which must be taken into account as early as possible in the design process. Electronic components must be adapted to the operating environment (temperature, humidity, vibrations, etc.) and thus, under certain circumstances, additional measures such as encapsulation or cooling may be necessary to ensure component reliability.

System (functional-physical) integration therefore requires a systematic design procedure and modeling support with computer tools to analyze the system integration. Thus, to achieve a high degree of integration, the compatibility of operating principles and solution elements must be verified, taking into account beneficial and disruptive functions, and the interfaces for further integration (rough sizing) must be formulated in the system design step. The possible incompatibilities must therefore be detected and eliminated during the preliminary system design and analysis. Otherwise, there may be costly changes during the specific-domains fine design.

In traditional mechanical engineering, the problem of system integration lies in a good definition of the part's shapes and their properties determining for a mechanical assembly design. In mechatronic system design, however, various interactions (e.g. behavior, electromagnetic compatibility, etc.) have to be taken into account to verify the incompatibilities.

Today, several multi-domain modeling and simulation software tools, such as Simcenter/Amesim, Matlab/Simscape, Modelica/Dymola, etc., allow the study of multi-domain and multi-physics interactions in a mechatronic system before embarking on a detailed geometric design, which is usually costly. These software tools provide a 0D/1D simulation suite for modeling and analyzing multi-domain systems and predicting their multi-disciplinary performance. At the lowest level, the dynamic behavior of the components is described by mathematical equations or algorithms. Domain-specific knowledge is represented by libraries containing components for mechanical, electrical, control, thermal, pneumatic, hydraulic, power train, thermodynamics, vehicle dynamics, air conditioning, etc.

Nevertheless, the multi-physics behavior is often very dependent on the geometry of the components and the geometric interfaces. This requires the development of compact models coupling simple geometry and physical behavior, such as vibration and thermal behavior, to analyze the location of components on the choice of a mechatronic system architecture.

In addition to the use of 0D/1D computer tools to simulate the interactions and multiphysics compatibilities, close cooperation between domain specific disciplines is necessary, as the evaluation of these interactions requires detailed technical knowledge.

1.1.2.3 Multiphysics Simulation (MPS)

When modeling multi-physics systems, each physics is represented, in general, as a system of nonlinear, time-dependent equations. These equations can be a combination of Partial Differential Equations (PDE), Ordinary Differential Equations (ODE) or Differential Algebraic Equations (DAE). Multiphysics applications require a way to exchange information between physical components. Indeed, multiphysics problems require the computation of interacting fields that may have different time and/or space scales and may also be defined over different spatial regions.

Space-dependent multiphysics systems can be modeled using a single-domain or multi-domain approach. For a single-domain approach, the coupled system is considered globally, and the state variables are defined over the entire spatial domain of the problem. A simple example for single-domain approach is a thermo-mechanical system whose solution consists of the temperature and displacements. For a multi-domain approach, different sets of state variables and/or models may be considered on different spatial or physics domains. A simple example for different spatial domains is Fluid Structure Interaction (FSI), where the fluid velocity and pressure are defined in the fluid domain and displacements are defined in the solid domain. Multi-domain coupling, must respect the mathematical formulation of the coupled problem and must account for different meshes and discretizations used in each domain [Keyes et al., 2013].

Time-dependent multi-physics systems can be modeled using a monolithic or partitioned approach. In a monolithic approach, the physics are naturally and intrinsically coupled. The system is represented by a single system of ODEs or DAEs, and the time integration is accomplished with a single solver; all quantities are determined simultaneously and the coupling between the state-variables in this case is strong. In a partitioned approach, the system is decomposed into a set of subsystems, each of which is represented by a partition that is then coupled to other subsystem partitions via partition input-output relationships. The strategies used to solve coupled sets of physics equations can be generally categorized as tight coupling (or strong partitioned coupling) and loose coupling. Figure 1.4 illustrates the different multiphysics couplings.

In tight coupling solution methods, a single system of equations is assembled and solved for the full set of coupled physics. The nonlinear iterations operate on the full system of equations simultaneously, taking into account the interactions between the equations for the coupled physics in each iteration. In cases where there is strong coupling between the physics, this approach can have faster convergence rates than loose coupling. The primary disadvantage of this approach is that it necessitates tighter coordination between the codes to solve the individual physics [Hales et al., 2015].

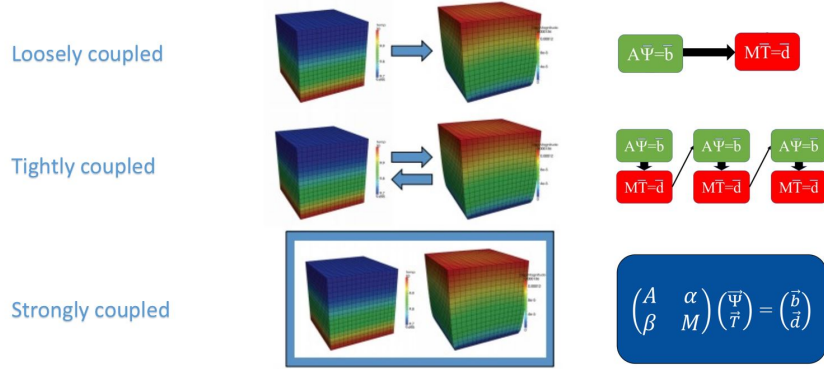


Figure 1.4: Multiphysics couplings

In loose coupling, the distinct physics in a coupled problem are solved individually, keeping the solutions for the other physics fixed. After a solution is obtained for an individual physics, it is transferred to other physics that depend on it, and solutions are obtained for those physics. More precisely, partitions in loosely coupled multiphysics system are time integrated in a conventional serial staggered procedure (sequential integration).

Loose coupling allows for the use of existing software modules such as dedicated fluid-dynamics and structural-dynamics software. Loosely coupled partitions can be coupled either explicitly or implicitly. Under explicit loose coupling, advancement of all partition solutions from time t to $t + \Delta t$ is accomplished based on shared information at t . Alternatively, implicit partition coupling is formulated such that one or more partitions require other-module data in $(t; t + \Delta t)$ before the solution can be advanced; the partitions are thus tied together in a linear or nonlinear system that must be solved for time advancement [Gasmi et al., 2013].

It is necessary here to distinguish between physical and numerical couplings. Indeed, the physical phenomena can be strongly or weakly coupled, from a physical point of view. The resolution methods on their side can be tightly or loosely coupled, from a numerical point of view. Nevertheless, a strong multiphysics coupling could be modeled and solved with a loose coupling. In the same way, a weak multiphysics coupling could be modeled and solved with a tight numerical coupling. There is therefore no one-to-one correlation between physical and numerical couplings. All depends on the physical phenomena and the numerical means of resolution.

Simulating complex multiphysics systems can be very expensive, and sometimes is even infeasible due to the wide range of scales. In particular, in the mechatronic design process, simulations must be run many times to perform parameter studies, design optimization or for system control design and analysis. In addition,

using full order models such as Finite Volume Method (FVM), Finite Difference Method (FDM) and Finite Element Method (FEM) causes to large algebraic systems and high computational times. It is therefore necessary to search for methodologies to exploit the potentials of multiphysics simulation and adapt them to the needs of mechatronic design. The use of compact 0D/1D models and reduced order models techniques are ways to reduce the computational complexity in such contexts.

Collaboration between multiphysics/mechatronics interdisciplinary groups is also one of the most important barriers to multiphysics applications for mechatronics system design. Indeed, the move to a multiphysics framework for mechatronics and thus to more complex simulations, will require the input of code and model developers to optimize efficiency, the input of mathematicians to study stability and accuracy, and the input of the mechatronics engineers to ensure the integration and the overall analysis of the mechatronic system.

It is thus necessary also to have interdisciplinary collaborative frameworks that allow for knowledge sharing and communication to understand the problems of simulation and the assumptions behind the underlying models. Without this input, it is likely impossible to perform a complete analysis or optimization of the mechatronic system, and time will be lost as assumptions or necessary physical conditions may be violated during development. All members of the interdisciplinary team must understand both their role and that of the other participants. The most successful collaborations will produce multiphysics simulations that yield new physical knowledge with computational efficiency and mathematical accuracy [Keyes et al., 2013].

1.1.2.4 Design for X (DFX)

Efforts to reduce total product life-cycle costs through design innovation have always been a critical issue of the manufacturing industry. As a result, researchers have focused on integrating development constraints, methods, and costs in the preliminary design phase of complex products. Hence the emergence of design for assembly (DFA), design for manufacture (DFM), design for manufacture and assembly (DFMA), design for disassembly (DFD), design for environment (DFE), design for recycling (DFR), design for quality (DFQ), design for life cycle (DFLC), etc. These design methodologies are often referred as design for X (DFX).

For instance, Design for Manufacture and Assembly (DFMA) is an engineering methodology that integrates Design for Manufacture (DFM) and Design for Assembly (DFA) [Boothroyd et al., 2010]. Beginning in the early design phases of the product life cycle, DFMA aims to reduce time-to-market and total production costs by prioritizing the ease of manufacturing product parts (DFM) and simplified assembly of parts (DFA) into the final product.

DFM is defined as "the set of policies, techniques, practices, and attitudes

that cause a product to be designed for optimal manufacturing cost, optimal manufacturing quality, and optimal life cycle support, serviceability, reliability, and recyclability" [Kuo and Zhang, 1995]. The objectives of the DFM approach are to: (1) identify product concepts that are inherently easy to manufacture, (2) focus on component design for ease of manufacture and assembly, and (3) integrate manufacturing process design and product design to ensure the best match between needs and requirements. DFM seeks to identify appropriate materials and manufacturing processes for the components considered in a product design, based on the combination of various product capabilities and limitations. Typically, DFM focuses on a particular manufacturing process, e.g., machining, stamping, injection molding, assembly, etc., and seeks to incorporate measures into the early design of the product that can prevent manufacturing problems and greatly simplify the production process [Kuo and Zhang, 1995].

Manufacturability is a DFM metric to measure how difficult or easy the design is, from manufacturing perspective. Similarly, producibility is a metric to measure how difficult or easy the design is, from production perspective. Therefore, the difference between the two metrics manufacturability and producibility is related to the difference between manufacturing and production. Indeed, manufacturing is generally defined as the process that involves the conversion of raw materials into finished products using machinery. Production, on the other hand, refers to the conversion of inputs, including raw materials and partially finished products, into finished products. Therefore, production is a broader term that encompasses the financial activities of a company as well as other non-tangible elements.

To illustrate the difference between producibility and manufacturability, let us consider the example of an Electronic Throttle Body (Electronic Throttle Body (ETB)), as illustrated in figure 1.5. This example will be used later in the contributions section as an example to validate some of our research work.

As a mechatronic system, the ETB includes mechanical parts, electrical power, electronic sensors and a control system. The ETB is used in automobiles with combustion engines to control the air supply to the internal combustion engine (ICE). Air enters through the air intake and is admitted into the engine through the ETB air outlet after being regulated by the throttle valve according to the control of the power control module (PCM). A mechanical fail-safe system is incorporated in the ETB so that the air flow to the internal combustion engine is not completely shut-off in case of a problem with the ETB or its control.

A company that produces ETBs may only need to manufacture the body. The other parts must be purchased and assembled. Thus, in this case of ETB production, the company can use the manufacturing for some parts, but not for all. Hence, producibility metric can be viewed as the aggregation of several metrics, including manufacturability and assemblability. Indeed, assembly processes are adapted to assess implementation of mechatronic products in conditions of high volume and mass production.

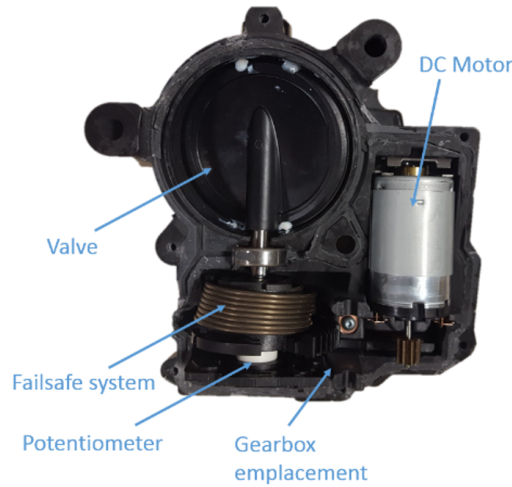


Figure 1.5: Electronic throttle body system

Assembly plays a fundamental role in the production of most engineered systems. Parts that have been individually formed or machined to meet designed specifications are assembled into a configuration that achieves the functions of the final product or mechanism. The economic importance of assembly as a production process has led to extensive efforts to improve the efficiency and cost effectiveness of assembly operations [Sanderson et al., 1990]. An assembly sequence plan is a high-level plan for building a product from its components. It indicates which sets of parts form sub-assemblies, the order in which the parts and sub-assemblies are to be inserted into each sub-assembly, and the nominal paths along which these insertions are to be made [Chakrabarty and Wolter, 1997]. Indeed, there are many possible assembly sequences for a product. Optimal sequence selection effectively reduces the cost and time of the assembly process [Kanai et al., 1996]. This method takes into account the shape of the parts, the connection typologies, the manual handling operations, as well as the calculation time and cost and the difficulties of assembly between parts. Assemblability is a DFA metric to evaluate the quality of a product design from the assembly producibility point of view [Ohashi et al., 2002]. Assemblability assessment allows to reduce the number of parts, to merge the parts when possible and to simplify manual, fixed automation, or programmable automation of systems assembly operations [Favi and Germani, 2012].

For automated assembly operations, Assembly Sequence Planning (Assembly Sequence Planning (ASP)) provides feedback to the designer on the suitability of the product for robot assembly. On the other hand, assembly planning is the basis for the automatic generation of robot programs. The requirements for the design of new robot types can also be obtained from the ASP, e.g. the number of Degree Of Freedom (DOF), joint angle ranges, etc [Thomas et al., 2003].

Products can be disassembled to allow maintenance, improve serviceability, and to meet end-of-life objectives such as product reuse, remanufacturing, and recycling. In this age of environmental awareness, end-of-life goals such as component reuse, remanufacturing, and recycling are some of the most important reasons to disassemble products [Desai and Mital, 2003]. As for assemblability, disassemblability is a Design for Disassembly (DFD) metric to evaluate the degree of easiness of disassembly. Disassemblability can be evaluated by measuring or calculating indicators such as the disassembly time, the effort and/or the energy/entropy required to accomplish the disassembly operation [Vanegas et al., 2018]. Automating the generation and the optimization of Disassembly Sequence Planning (Disassembly Sequence Planning (DSP)) is one of main challenge issues for disassemblability assessment.

ASP and DSP activities are proven to be NP-hard and thus their effective and efficient solutions have been a challenge for the researchers in this field. Indeed, ASP and DSP of a complex product easily runs into the combinatorial explosion, as the number of assembly/disassembly sequences is exponentially proportional to the number of parts or components in the product. Thus, several soft computing/metaheuristic algorithms were developed to solve the problem, such as Simulated Annealing [Motavalli and Islam, 1997, Hong and Cho, 1999], Artificial Immune System [Cao and Xiao, 2007], Genetic Algorithms, Ant Colony Optimization [Wang et al., 2005, Guo et al., 2007], Artificial Neural Networks [Chen, 1992, Hong and Cho, 1995, Sinanoğlu and Börklü, 2005, Chen et al., 2008], Particle Swarm Optimization [Zhang et al., 2014, Lv and Lu, 2010, Wang and Liu, 2010, Tseng et al., 2011]. In most of these research works the assembly was represented either with graphs and diagrams [De Fazio and Whitney, 1987, De Mello and Sanderson, 1991, Santochi and Dini, 1992, Gu et al., 2008] or with 3D CAD models [Hadj et al., 2018, Said et al., 2019, Bedeoui et al., 2019, Belhadj et al., 2020, Belhadj et al., 2021].

Figure 1.6 illustrates an example of DSP generation and quality assessment based on CAD data.

However, most of previous research works focused on detailed product geometry, which essentially introduces combinatorial problems in an ASP-DSP phase due to the product design complexity. Besides, traditional well-known DFA/DFD approaches can be considered as reactive, since they work on a detailed product geometry after all design decisions made, hence they lead to redesigns and delays due to late decisions changes. These approaches therefore result in late decision support and missing a true opportunity for an efficient product development process [Demoly et al., 2011]. Therefore, more systematic approaches to the analysis, design, and planning of the assembly/disassembly operations or systems are needed to enhance their performance and enable their cost-effective implementation. In

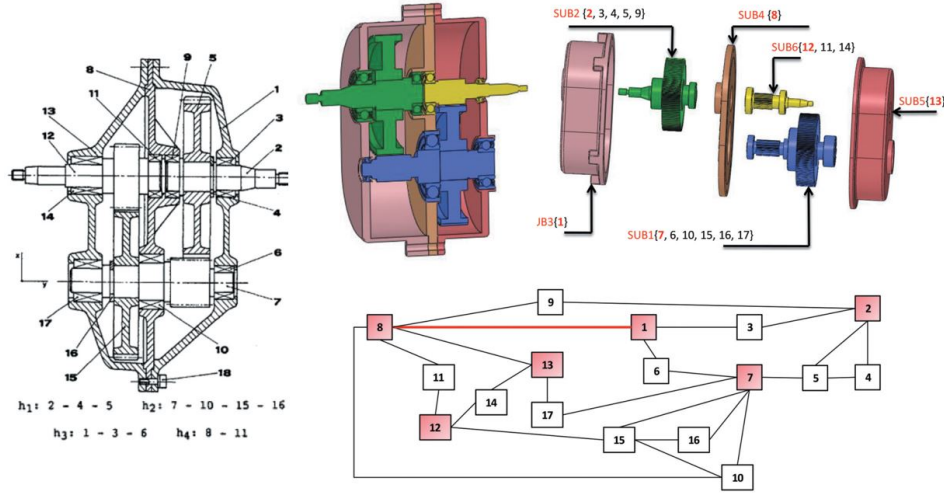


Figure 1.6: Generation of disassembly plans and quality assessment based on CAD data [Belhadj et al., 2020]

particular, it is necessary to define the ASP-DSP in the preliminary stages of design by introducing and applying knowledge about the assembly/disassembly process in order to provide knowledge about the product/production context to support the systems engineering development process, especially for product structuring. Thus, there is value in integrating assembly/disassembly process engineering information and knowledge into the early phases of the product development process in a top-down, proactive manner to provide designers with an MBSE-based assembly context. The definition of the product and its assembly/disassembly sequence requires both the improvement and the complete understanding of the relationships between the different components of the product and the associated assembly logic. Therefore, a new modeling approach is required to integrate the views of the different stakeholders involved in the early stages of the product life cycle.

1.1.2.5 Multidisciplinary Design Analysis and Optimization (MDAO)

For the design of complex engineering systems, multidisciplinary models are used in the design cycle with a strong interaction between them. Therefore, optimizing a model without taking into account the other models will generate integration problems in the final product. The optimal design may even tend to converge on an absurd solution for the system [Chapman and Pinfold, 2001]. Given this state, Multidisciplinary Design Analysis and Optimization (MDAO) has been recognized as a promising solution to optimize globally the complex engineering systems [Alexandrov, 2005].

Multidisciplinary Design Analysis and Optimization (MDAO) or , for short, Multidisciplinary Design Optimization (MDO) is “a methodology for the design of

complex engineering systems that are governed by mutually interacting physical phenomena and made up of distinct interacting subsystems in their design, everything influences everything else” [Sobieszcanski-Sobieski, 1995].

To express the MDO/MDAO problems, the basic formulation is the All-At-Once formulation (AAO), as defined in the Figure 1.7. This formulation contains all the coupling variables, their copies, state variables, consistency constraints and residual equations. Other formulations are derived from this definition. For the details about this formulation, readers can refer to [Cramer et al., 1994].

The term MDO architecture identifies how the simulation blocks, analysis ele-

$$\begin{aligned}
& \text{minimize} && f_0(x, y) + \sum_{i=1}^N f_i(x_0, x_i, y_i) \\
& \text{with respect to} && x, \hat{y}, y, \bar{y} \\
& \text{subject to} && c_0(x, y) \geq 0 \\
& && c_i(x_0, x_i, y_i) \geq 0 && \text{for } i = 1, \dots, N \\
& && c_i^c = \hat{y}_i - y_i = 0 && \text{for } i = 1, \dots, N \\
& && \mathcal{R}_i(x_0, x_i, \hat{y}_{j \neq i}, \bar{y}_i, y_i) = 0 && \text{for } i = 1, \dots, N.
\end{aligned}$$

x : design variables.
 x_0 : shared design variables.
 x_i : discipline design variables.
 y : coupling variables.
 \bar{y} : state variables
 \hat{y} : independant copies of variables (target)
 f_0 : shared objective function
 f_i : discipline objective functions
 c : design constraints
 c_0 : shared constraints
 c_i : discipline constraints
 c_i^c : consistency constraints
 \mathcal{R}_i : Governing equations in residual form

Figure 1.7: All-At-Once formulation (AAO) MDO problem formulation

ments, algorithms and overall process flows are related between each other. The architecture refers to the algorithmic strategy to solve the problem. Different MDO architectures are available in the literature but they are presented in different manners. An interesting survey is provided by [Martins and Lambe, 2013] where clear algorithms of these architectures are proposed to basically describe MDO architectures in unified diagrams called eXtended Design Structure Matrix (XDSM).

To read the example of XDSM diagram presented in Figure 1.8 correctly, we have to follow the indicated numbers step by step. The directions of exchanging the parameters are also illustrated in the Figure 1.8. The components consist of the discipline analyses represented by rectangles and a special component (driver) that controls the iteration and are represented by a rounded rectangle. The data flow is shown as thick gray lines. The components take data inputs from the vertical direction and output data in the horizontal direction. Thus, the connections above the diagonal flow from left to right and top to bottom, and the connections below the diagonal flow from right to left and bottom to top. The off-diagonal nodes in the shape of parallelograms are used to label the data. External inputs and outputs are placed on the outer edges of the diagram. The thin black lines show the driver process flow. Loops are represented by $j \rightarrow k$ with $j < k$. It means that if in the j step convergence condition is not reached we continue the process from the k step.

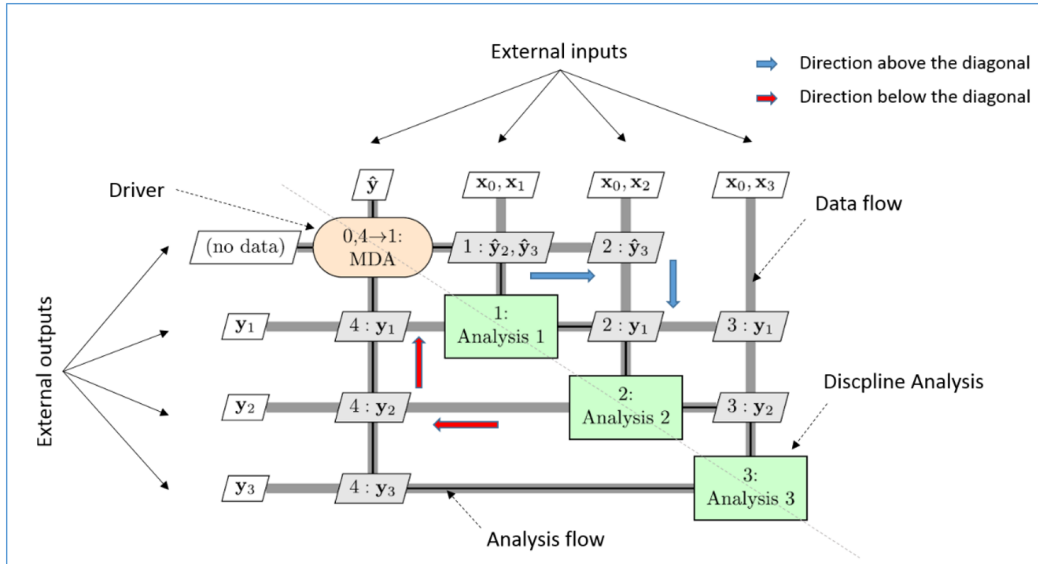


Figure 1.8: XDSM diagram Block Gauss-Seidel multidisciplinary analysis (MDA) process to solve a three-discipline coupled system

Two kinds of architectures can be found in the MDO/MDAO literature [Tedford and Martins, 2010]:

- Single-level architectures : The whole system is optimized using a single optimization process:
 - All-At-Once (AAO)
 - Multidisciplinary Feasible (MDF)
 - Individual Discipline Feasible (IDF)
- Multilevel architectures - Mono-disciplinary optimizations are first conducted, before optimizing the whole system:

- Collaborative Optimization (CO)
- Concurrent SubSpace Optimization (CSSO)
- Bi-Level Integrated Systems Synthesis (BLISS)
- Analytical Target Cascading (ATC)

Unlike traditional optimization methods, **MDO** is a multidisciplinary methodology (data analysis, visualization, sensitivity analysis, optimization architecture...). **MDO** can be performed with Process Integration and Design Optimization (PIDO) tools. They are specialized in **MDO/MDAO** workflow management and offer various features to integrate external tools and analysis models in a common framework. Some examples of PIDOs are illustrated in Figure 1.9 (iSIGHT, modeFRONTIER, PAnO, ModelCenter, Optimus). PIDO tools suggest relatively

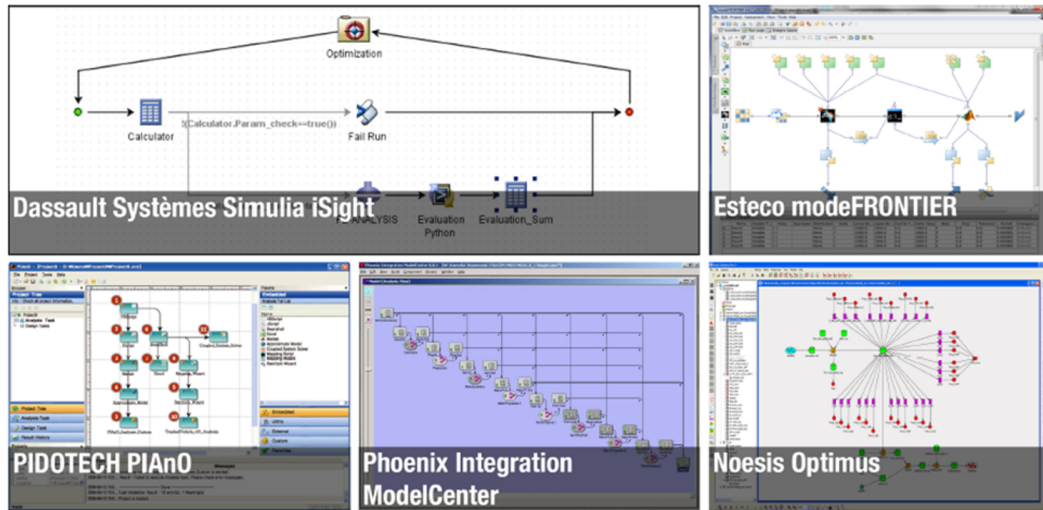


Figure 1.9: Examples of Process Integration and Design Optimization (PIDO) tools

the same functionalities: Optimization problem formulation, creating links between analysis models involved in the **MDO** workflow and finally running the optimization algorithm and tools for result analysis.

In all the engineering domains, designers use computationally expensive numerical models. The optimization process based on the integration of such models can take long time due to the significant number of iterations. Surrogate modeling is used in this case to make an approximation of the expensive model and then use it in the optimization process.

In the field of surrogate modeling, three steps are required as we can see in Figure 1.10. An initial set of sample points is generated using a statistical method of Design of Experiment (DoE). These points are then used as inputs to run the simulation model and get function evaluations. Finally, a surrogate model type is

selected to represent the response surface of the simulation model (Gaussian process (GP), polynomial regression, multi-variate adaptive regression splines (MARS)). If the precision of the model is not sufficient, new points are generated to adapt the model [Barton and Meckesheimer, 2006].

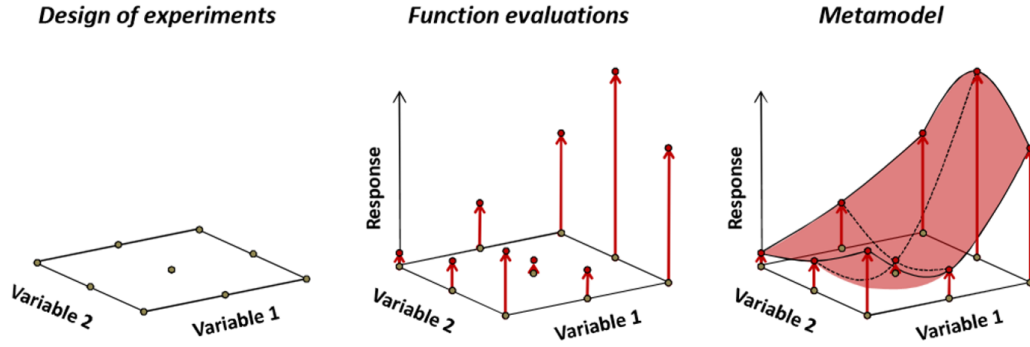


Figure 1.10: Surrogate modeling process

Another Surrogate modeling method, called Space Mapping, is relatively different from the classical methods mentioned above. The difference is that we assume the existence of two models for the same system: a fine model (precise and expensive) and a coarse model (cheaper and less accurate). The idea is to use the coarse model in the optimization process and update it with the fine model to improve the accuracy (Figure 1.11). The algorithm establishes a mapping between the two models using Broyden updates [Bandler et al., 1994]. The critical part of the process is the mapping function. Many techniques exist depending on the mapping function (Input Space Mapping, Manifold Space Mapping, Aggressive Space Mapping, Output Space Mapping, etc).

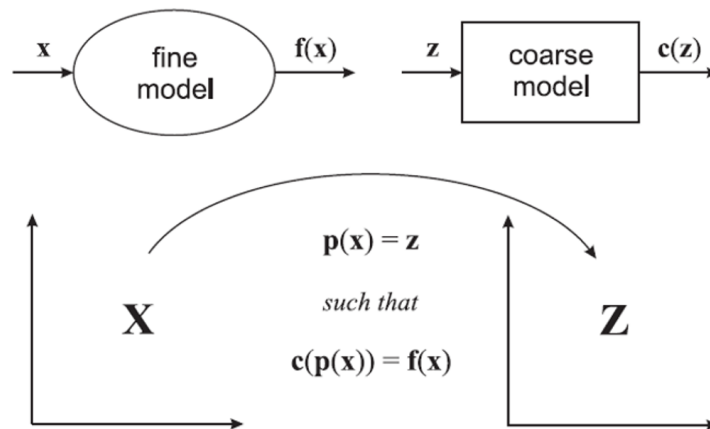


Figure 1.11: Space mapping concept

As indicated in this section, the workflow of analysis and optimization of the design process of a complex system is specified with an XDSM diagram for MDO specialists. However, the trend today is aligning people, processes, and technology around a single vision of a product promised by MBSE. Indeed, the use of a system architecture model as a single source of truth for describing the evolving system design throughout its product life-cycle reduces the development cost and risk of complex systems. The issue is the disconnection between systems engineers and engineering analysts that prevents MBSE from achieving its full potential.

1.1.2.6 Interdisciplinary Collaborative Design and Knowledge Management (ICDKM)

To improve interoperability between disciplinary engineering tools, many product models are created to support the Product Lifecycle Management (PLM) approach through data and information management (Function-Behavior-Structure, Core Product Model, Product Process Organization, etc.). However, these models do not focus on the knowledge management built into Computer Aided Design (CAD) and Computer Aided Engineering (CAE) models. In this context, Knowledge Configuration Management (KCM) has been proposed to manage knowledge encapsulated in disciplinary models. It ensures the consistency of parameters through several product design and simulation activities in a context of collaborative engineering using the concept of Configuration Management (CM).

CM is a managerial discipline that aims to ensure consistency and accuracy in product knowledge throughout its life cycle and for the same purpose, it is used to varying degrees in most organizations [Niknam and Ovtcharova, 2013].

CM is defined in ISO 10007: 2003 releases as:

- Configuration: interrelated functional and physical characteristics of a product defined in the product configuration information.
- Product configuration information: requirements for the design, implementation, verification and support of a product.

Product configuration ensures the consistency through the design process by managing configuration information organized into “Configuration Items”. All the configuration items must be coherent over all the design process following the evolution into an iterative process and keep traceability of any changes. Configuration management can be particularly interesting to manage parameters and rules used in several CAD/CAE models.

The first objective of KCM is to enable designers to use parameters consistently in collaborative design. In fact, the lack of communication among different steps of the design cycle causes consistency problems in parameters. In Concurrent Engineering (CE), all participants need to access all relevant up-to-date product

information. This methodology is based on the concept of crucial knowledge to keep the design models consistent with each other [Monticolo et al., 2015]. Crucial knowledge refers to the sufficient and necessary knowledge that needs to be shared for the activity of a company [Saad and Chakhar, 2009]. Here, the knowledge consists of contextualized parameters and rules that are critical for the collaborative product development. Therefore, this crucial knowledge is structured and has its own lifecycle according to the Global Analysis Methodology (GAMETH) framework [Grundstein and Rosenthal-Sabroux, 2004]. This framework defines a cycle with four facets which are: identify, formalize, value and update.

The principle is the capitalization of critical knowledge extracted from different expert models, into an abstract generic information entity called an Information Core Entity (ICE). The ICEs are grouped in a Knowledge Configuration (KC). Then each stakeholder instantiates the necessary ICEs in his Usage Configuration (UC). This principle is illustrated in Figure 1.12.

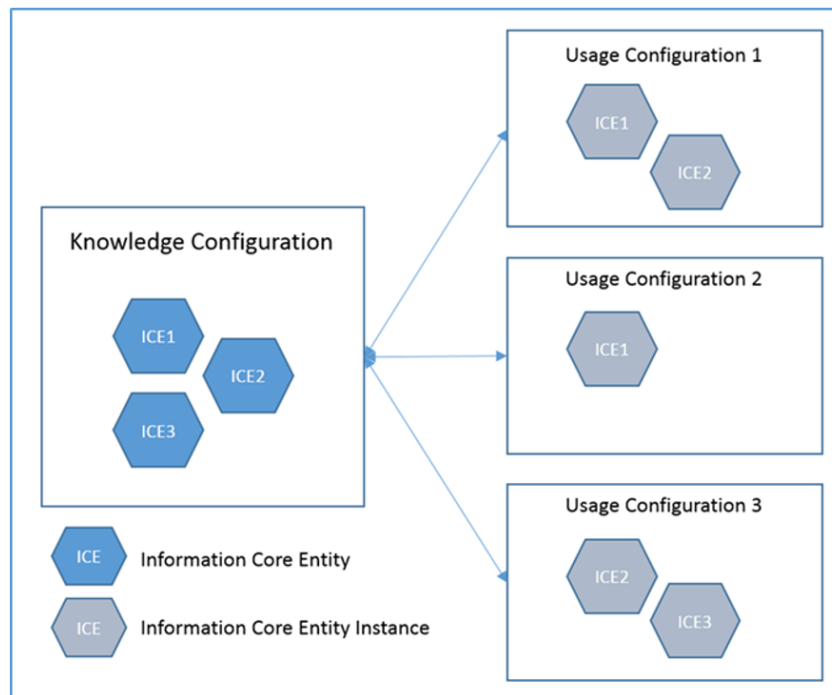


Figure 1.12: Knowledge Configuration Model principle

Main concepts of knowledge configuration:

- *Information Core Entity (ICE)*: ICE is an indecomposable generic entity that can capitalize crucial data extracted from disciplinary models in order to move from the data state to the information state. It is composed of parameters and generic constraints. These constraints include:

- Mathematic relations

- Expert rules (if ... then ...)
- Boundary conditions (min, max, default)
- Discrete values table

The creation of **ICEs** results from an analysis to identify data and information crucial to capitalize and share in the design process, especially between disciplinary models. The set of **ICEs** forms a dynamic knowledge base which is enriched as the project evolves.

- *Usage Configuration (UC)*: Each user makes a selection in the **ICE** database to retain only those he needs for his design activity. Then, the user instantiates this selection in a **UC** and synchronizes it with a disciplinary model. Thus, a **UC** is constituted of **ICE** instances and represents the knowledge encapsulated in a disciplinary model. The notion of configuration refers to all the services of version management, management changes and consistency management in the **UC**. Therefore, in a **UC**, a user can backup multiple versions based on changing parameter values or adding and deleting instances of **ICEs**. The interest of configurations is to have a mode of homogeneous representation of knowledge that can be used in different activities of the design process. It is through the **UCs** that the coherence of shared knowledge is ensured because it is very difficult to directly compare several disciplinary models implemented in different heterogeneous tools.

- *Knowledge Configuration (KC)*: **KC** contains the **ICEs** and **UCs** that will be used in the collaboration. It can be considered as the dashboard of the collaboration. **UCs** are compared in **KC** to give users information on the existence of possible conflicts. This global configuration ensures the management and collaboration of all the **UCs** and represents all the activities of a project milestone. **ICEs** in **KC** are either directly instantiated from the generic knowledge base or recovered from other previous configurations. When the collaboration starts, the project manager can visualize the existing conflicts between the different instances of **ICEs** of the published **UCs**.

- *KCM metamodel*: The **KCM UML** metamodel is shown in Figure 1.13. The figure shows the links between the different concepts discussed above.

Particularly, **KCM** manages:

- Technical data: the parameters and expert rules extracted from disciplinary models.
- Information: the data identified, structured and organized into a specific entity to construct a technical and generic product information baseline.

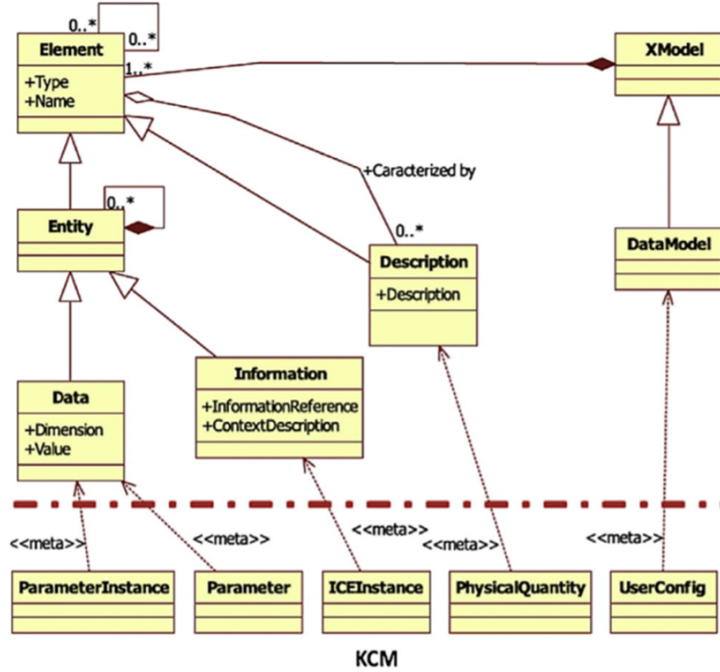


Figure 1.13: UML meta-model of KCM [Monticolo et al., 2015]

- **Product Knowledge**: a set of technical product information entities instantiated from the baseline in a configuration used in specific design or simulation activity. This configuration is synchronized with a specific **CAD** or **CAE** model.

- *KCM use*: Each user, working on very different expert models (design and simulation domain, different tools, different components, etc.), exports data from multiple product architectures into their configurations simultaneously. The consistency management between configurations is a main utility of **KCM**, because users can be warned if conflicts occur between data shared by several expert models.

Two kinds of consistency checking can be considered:

- Consistency in **UC** level: parameters must respect all the rules defined in the **UC**
- Consistency in **KC** level: Instances of the same **ICE** must contain the same values.

1.1.3 Positioning

My thesis, defended in January 2012, focused on a contribution to the integration of multiphysics simulation in the design process of mechatronic systems [Hammadi, 2012]. Following this, I continued to work on the development of

methodologies for mechatronic integrated design that meets the Mechatronics objectives, as fixed in the definition NF E01-010 introduced in section 1.1.1.

As previously mentioned, this work is divided into three research activities:

- MBSE-based approaches for the integrated design of mechatronic systems.
- Multiphysics simulation methods for mechatronics design.
- Interdisciplinary collaborative design of mechatronic systems.

This research work is consistent with the research activities of my research team IS2M: Ingénierie des Systèmes Mécatroniques et Multiphysiques, for Mechatronics and Multiphysics Systems Engineering, at Quartz EA7393 laboratory.

The objective of the first and second research activities is to ensure the synergy and integration of the mechatronic design approach, by adapting the MBSE/SysML method to integrate activities and knowledge related to the MPS, MDAO and DFX domains. The figure 1.14 gives an overview of the areas covered by our research on the development of integrated mechatronic design approaches.

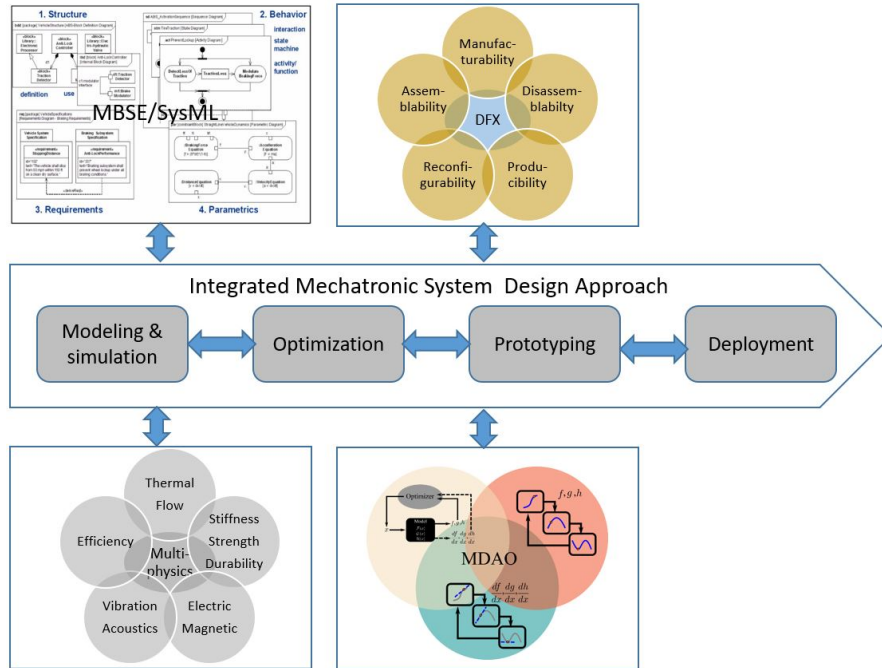


Figure 1.14: Integrated mechatronic system design approach

In the literature, we can find works that have dealt with this theme. For instance, [van Amerongen, 2003] discussed the need for an integrated approach when designing the mechanical and control parts of a mechatronic system. Mechatronic design requires models that, in addition to the controller parameters, retain all relevant physical parameters of the construction itself. Likewise, [Cabrera et al., 2010]

identified a set of challenges related to the design of mechatronic systems. The challenges are mostly related to integration of design and analysis tools, and automation of current design practices. Addressing these challenges enables the adoption of a concurrent development approach in which the synergetic effects that characterize mechatronic systems can be taken into account during design. The main argument is that there is a need to consider the complex interdependencies among subsystems and the designers that develop them. In addition, [Hehenberger et al., 2010] presented an approach for consistency checking of mechatronic design models based on the development of specific-domain checking rules. However, the granularity of modeling information affects the writing of consistency rules. So that, the approach is limited for consistency rules of elements at the same level of granularity (same level of abstraction). Furthermore, [Mørkeberg Torry-Smith et al., 2013] believed that a common language is essential in developing mechatronics, and should be evaluated based on: its capability to represent the desired views effectively, its potential to be understood by engineers from the various domains, and its effect on the efficiency of the development process. In the same way, [Thramboulidis, 2013] addressed the main challenges in mechatronic system development including synergistic integration, size and complexity, reuse, as well as requirements handling and traceability, support for decision making, and maintaining consistency. The authors proposed SysML-based approach to address these challenges. Moreover, [Budinger et al., 2014] presented a metamodel form and associated construction procedure adapted to the sizing tasks of mechatronics systems. The scaling-law-based metamodels have the advantage of a compact form with a good predictive accuracy. More recently, [Ma et al., 2019] treated the integrated mechatronic design problem as a controller optimization problem with structural constraints. The design problem is reformulated as minimizing the H2-norm upper bound of the closed-loop system transmittance from the exogenous disturbance to the regulated variables over the intersection of convex and non-convex domains.

On our side, we were interested in proposing methodologies for integrated design of mechatronic systems based on SysML language. We have found it useful to integrate the DFX metrics, namely manufacturability, assemblability, disassemblability, reconfigurability, producibility, etc. in the preliminary design process of mechatronic systems. To evaluate and optimize the mechatronic design, it was also deemed necessary to integrate the possibilities of multiphysics simulation and multidisciplinary optimization.

For the third research activity, our objective is to develop an interdisciplinary collaboration framework allowing the management of knowledge shared between the different design domains: MBSE, MPS, MDAO and DFX. Figure 1.15 shows an overview on the specific domain knowledge that the mechatronics development engineer must share with other design members.

Research to improve the effectiveness and efficiency of product develop-

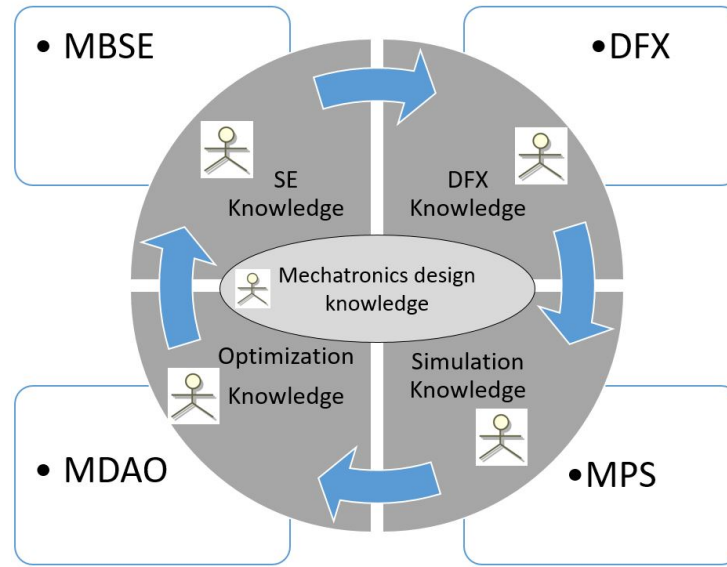


Figure 1.15: Interdisciplinary collaborative mechatronics design

ment based on knowledge creation and information transfer dates back to the early 1990s [Sanchez and Mahoney, 1996]. For mechatronic products design, [Dawson and Duffy, 1999] introduced the need for engineering knowledge management tools for storing past solutions and expert knowledge for the design of automatic precision machinery. Similarly, [Larses and Adamsson, 2004] discussed the model based development from knowledge management perspective. Thus, a model of competence integration for the design of mechatronic systems was derived. The model describes how integration can be performed; it also supplies a dichotomy of two explicit strategies and a set of drivers for a given strategy. [Karayel et al., 2004] applied a knowledge management approach to mechatronic systems. Indeed, a knowledge management model of an internet-based CNC machining center has been developed and applied. [Gardan and Matta, 2017] developed new models into a SysML diagram to answer knowledge management needs for complex systems engineering. Therefore, the requirements satisfaction and decision-making process can be performed using new SysML diagrams modeling the organization, coordination and decision making. More recently, [Yang et al., 2021] proposed a knowledge management approach to support an intelligent question answering scenario when implementing MBSE in system lifecycle. Authors make use of the Graph, Object, Point, Property, Role, and Relationship (GOPRR) approach to support MBSE formalisms which are transformed to knowledge graph models.

For our part, we believe that knowledge capitalization for mechatronic design is at the crossroads of knowledge sharing between mechatronics design domain and engineering disciplines related to MBSE, DFX, MPS and MDAO. For this reason,

it is necessary to develop collaboration tools that not only allow the mechatronics engineer to collaborate well with other specialists from different engineering disciplines, but also allow them to coordinate their activities with the specific needs of mechatronics design. Indeed, knowledge sharing is not limited to the knowledge models to be shared, but also to the collaborative process to solve multidisciplinary design problems in a collaborative manner.

1.1.4 Research issues

In the previous section, I presented our three research themes related to integrated and collaborative design of mechatronic systems. However, several issues have been identified that need to be addressed to achieve the research objectives. These issues can be summarized by the three research questions below:

- Issue 1: How can we integrate and evaluate the [DFX](#) metrics at the preliminary phases of mechatronic design using [MBSE](#)?
- Issue 2: How can we integrate compact models for multiphysics simulation to verify mechatronics design?
- Issue 3: How can we manage the knowledge for effective and efficient collaboration between the mechatronic system engineer and other engineers from specific engineering domains related to [MDAO](#), multiphysics simulation and manufacturing?

In the following sections of this chapter, I will detail our contributions to address the above issues.

1.2 MBSE-based approaches for the integrated design of mechatronic systems

The integrated design of a mechatronic system requires the development of an approach that allows, on the one hand, the simultaneous development of product models and modeling processes of the mechatronic system, and on the other hand, the integration of [DFX](#) metrics and their evaluation. Thus, this part of the contribution will be divided into two sub-sections:

- Integrating product model with production process model for mechatronics design.
- [MBSE](#) approaches to integrate [DFX](#) for mechatronics design

1.2.1 Production process modeling for mechatronic product design

The goal of a mechatronic design is not only to develop a product that meets the requested functionalities and the desired performance requirements, but also a product that can be manufactured with the available means, which are generally difficult

to modify. However, for the design of innovative mechatronic products, it is often necessary to adapt the means of production to achieve innovative design solutions. Moreover, modern production systems are more complex and sophisticated offering more production possibilities with a capacity of adaptation of the production system with the products to be realized by taking advantage of their reconfigurability. Thus, it is necessary to model the production system together with the mechatronic product in the upstream phases of design of the mechatronic product.

The objective of this section is to address the following two issues:

- How can the production system be jointly modeled with the mechatronic product?
- How can we develop a simulation model to verify the reconfigurability of a reconfigurable manufacturing system?

These two questions are the subject of the following two contributions.

1.2.1.1 Co-joint modeling of the mechatronic product with the manufacturing process

Background

The link between the development of a product and its production process is a key issue for the development of mechatronic systems. It is a common subject for the members of the Ingénierie des Systèmes Mécatroniques et Multiphysiques (IS2M) team. Research on integrated product-process design approaches has been considered since the 1990s. Several approaches and frameworks for integrated product-process development have been considered by researchers. Most research has approached the problem from two perspectives: (i) how to decompose the complex product into disjointed subsets, and (ii) how the design team constructs the decision making schedule, based on the dependencies between the subsets and the solutions studied for each subset.

Several works, related to systems engineering, have considered the first problem of decomposition of the product into disjoint subsets. To this end, a number of authors have proposed SysML as a language supporting the systems engineering methodologies they have suggested. For example, SysML4Mechatronics is an approach that uses the port concept of SysML to represent and analyze the influences of change in mechatronic production plants [Kernschmidt 2013]. In addition, [Mhenni et al. 2018] proposed an approach for the joint design of mechatronic products with their production systems. The proposed system engineering approach is based on SysML and it allows taking into account the interdependence between the mechatronic product and its production system.

Several other research works have addressed the problem of how the design team constructs the decision making schedule during the design process. Most of these

works were based on the Set-Based Concurrent Engineering (**SBCE**) (set-based concurrent engineering) method. The **SBCE** method starts by establishing a large set of possible solutions and gradually narrows this set of possibilities to converge on a final solution [Sobek 1999]. Design members examine, develop, and communicate the set of solutions in parallel and independently. As the design progresses, they progressively narrow their respective solution sets based on additional information provided by the development and design stakeholders. As designs converge, participants agree to stay within the boundaries of the sets, except in extreme circumstances, so that others can rely on their communication. Compared to working with one idea at a time, **SBCE** embraces the principle that processing sets of ideas leads to more robust and overall efficient optimized systems. Because **SBCE** can be performed without feedback or rework, development time and costs can be significantly reduced. However, there are several problems with implementing the **SBCE** approach in companies. A first problem is related to how companies determine how many design ideas to pursue, an important step in the early design process. Another problem is how firms organize themselves for the overall design by assigning teams to develop design ideas and ultimately design an architectural system [Ammar et al.].

Proposed approach

The systems engineering design process is illustrated in Figure 1.16. System architects propose concepts that meet the functional requirements of the product to be developed. The **SysML** modeling language is proposed to specify these concepts. Therefore, **SysML** is used to specify all traceability links and attributions between the developed system models. In addition, **SysML** is used to specify the appropriate manufacturing capabilities for the product design features. The company's manufacturing knowledge can be formalized as a set of **SysML** diagrams by the manufacturing engineers. In this way, the initial manufacturing constraints of the production system are identified and the first set of production system requirements are defined.

Figure 1.17 shows how the design team builds the decision making schedule, based on the **SBCE** method.

As the design of the mechatronic product evolves to define the preliminary architecture, taking into account the results of the conceptual and functional development of the product and its production system, the development process of the production system design also evolves to define the possible design solutions.

The mapping of the design space includes the definition of the different alternatives of each subsystem of the mechatronic product and the possible production solutions. The resulting set of possibilities is communicated to the other design collaborators to incorporate other specific design constraints (**DFX**), such as cost, quality, manufacturability, maintainability, safety, etc. To get closer to detailed system design, assemblies are integrated in an inter-sectional manner by seeking the intersection of feasible assemblies, imposing minimal constraints, and seeking design robustness. During this design process, the principle of inter-sectional integration is managed by

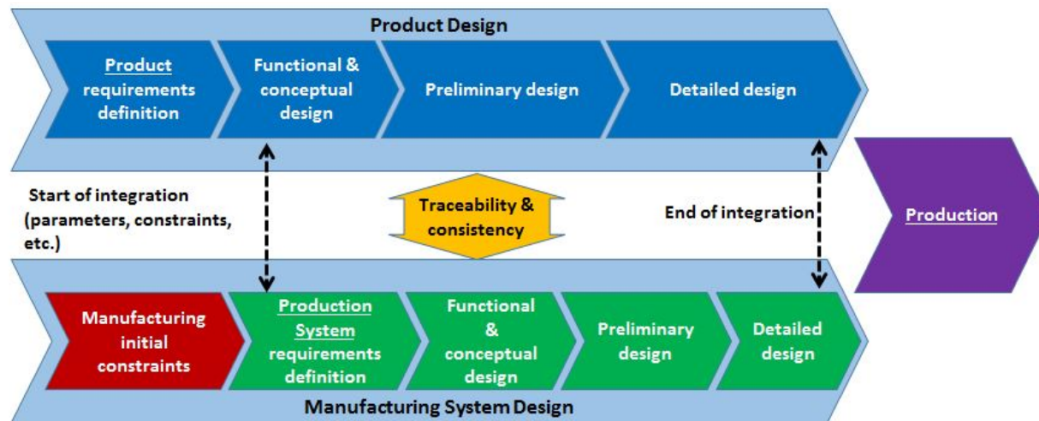


Figure 1.16: Systems engineering process for the joint design of a mechatronic product and its production system

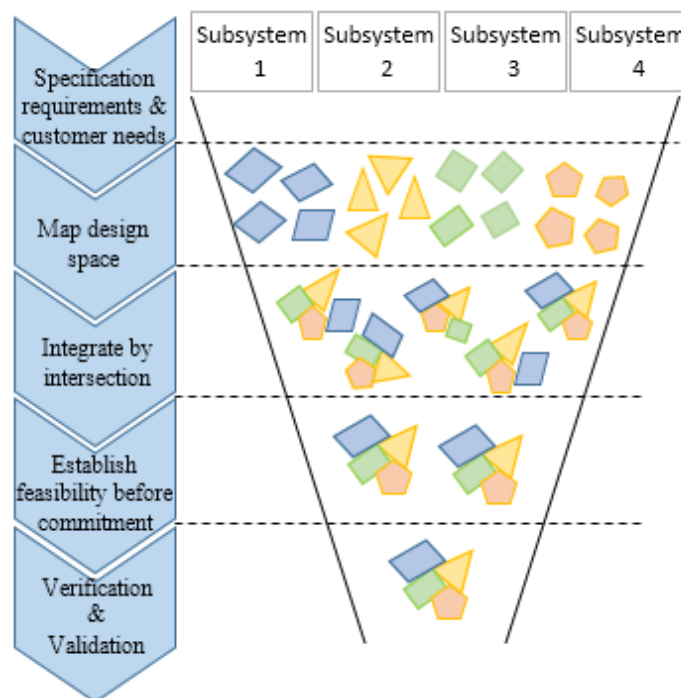


Figure 1.17: SBCE steps for concurrent engineering

identifying interactions between design solutions to eliminate infeasible alternatives.

For the evaluation of **DFX** metrics, mathematical evaluation models and algorithms to reduce the design space are required. The integration and evaluation of the **DFX** metrics will be detailed in the dedicated contribution.

Validation of the proposed approach: Case-study of an electronic throttle body

The proposed approach was applied to the case study of an electronic throttle body (**ETB**). Figure 1.18 shows the components of the **ETB**.

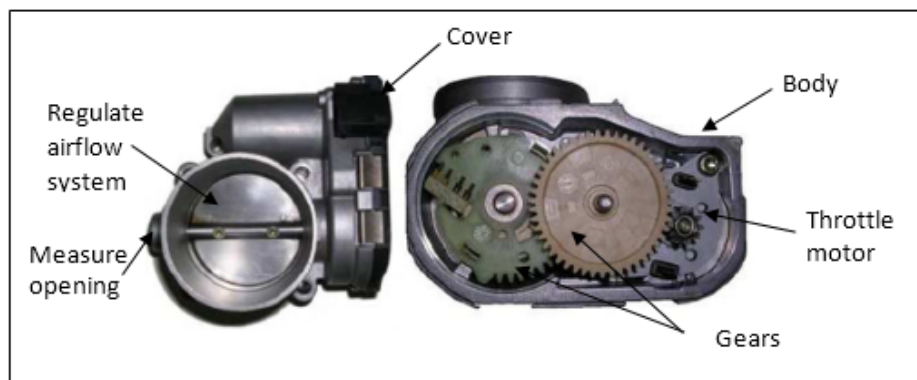


Figure 1.18: Electronic Throttle Body (ETB) components

As explained in section 1.1.2.4, the **ETB** is used in automobiles with combustion engines to control the air supply to the internal combustion engine (ICE). Figure 1.19 shows the functional architecture model of the **ETB** system developed using SysML.

The functional decomposition can be performed at different hierarchical levels until the subsystems and components needed to perform the different functions can be identified. In this case, we have identified 4 subsystems and 8 components as shown in Figure 1.20. The first subsystem is composed of the body (1), the cover (7), the air flow control system (3), the gearing system to adapt the mechanical energy (5) and the springs (4). The second subsystem is the actuator (2). The third subsystem is the sensing system (6) to measure the opening/closing angle, and the last subsystem is the controller (8).

The feasible design region, defined as the set of possible solutions, is determined by selecting the possible alternatives for each subsystem or component based on its properties. For example, the possible alternatives for the body (1) can be obtained by considering different possibilities for its properties, such as geometry, material and manufacturing process. In this case, the geometry of the **ETB** body can consist of one block, two blocks or three blocks. The material of the body may be metal or plastic, and in the case of a plastic body, it may consist of one plastic or two plastics (bi-plastic). The manufacturing process of the body can be molding or

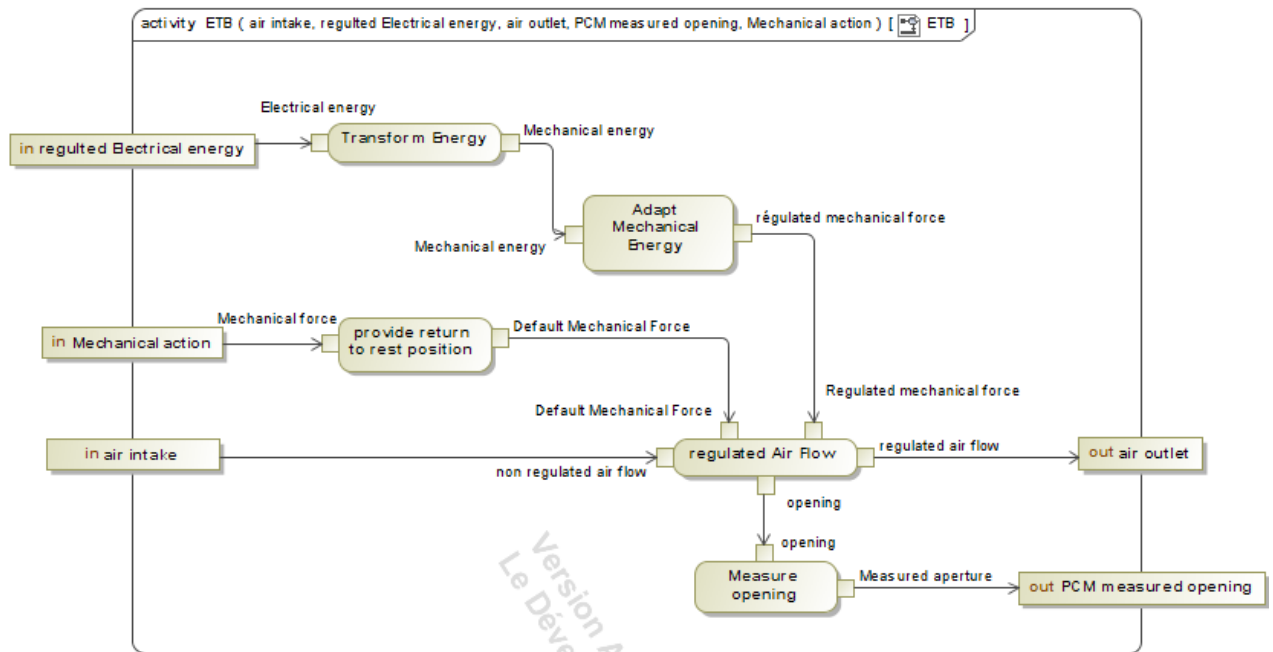


Figure 1.19: Functional architecture model of the ETB system developed with SysML

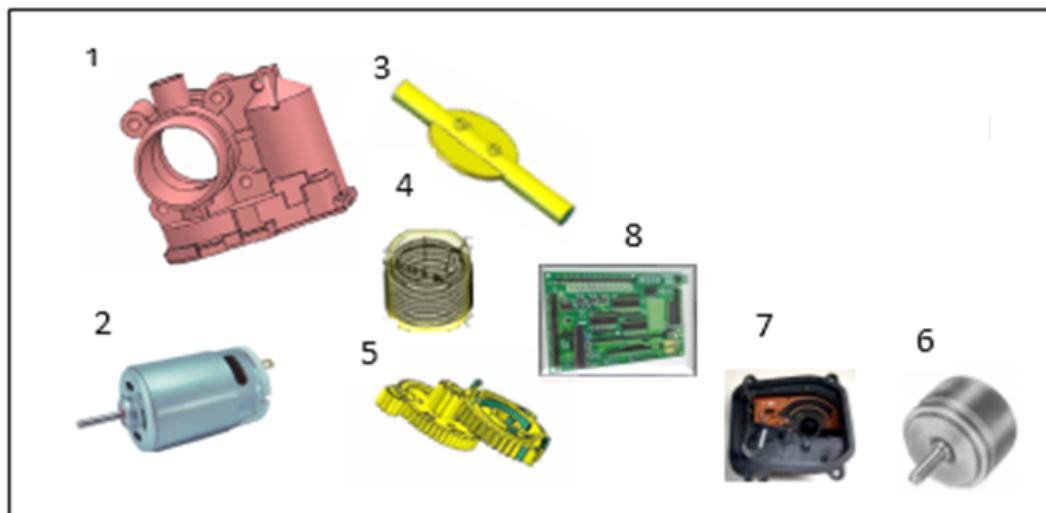


Figure 1.20: The ETB components

co-molding (multi-material injection molding). By similar reasoning with the other subsystems and components, we can define the design space of the **ETB**. Figure 1.21 shows 5 possible solutions for the body, 3 solutions for the airflow controller, 2 solutions for the gear system, 2 solutions for the spring component, 2 solutions for the actuator (motors), and 2 types of controllers. By combining these solutions, 480 potential **ETB** configurations can be obtained.

Using **SysML**, it is possible to represent the design space with a block definition diagram (BDD) as shown in Figure 1.22.

By looking at the different **ETB** components that can be produced by manufacturing, we can determine the different manufacturing processes and the machines that match with them. Using the same approach as for the **ETB** components, a **SysML** diagram can be drawn up containing the different alternatives for producing the **ETB** components. Figure 1.23 shows a **SysML** diagram representing the design space for choosing production processes and machines.

Once the two design spaces of the product and the production process have been identified, the next step in the **SBCE** method is to apply the principle of integration by intersection. The number of solutions is first reduced by eliminating those outside the intersection. Then, by adding additional constraints and requirements, the sets are reduced more and more until only one solution remains for the **ETB** as a product and for its production process. Figure 1.24 shows the **SysML** model of the final production process of the **ETB**.

Having defined the traceability links between the objects in the model relating to the product and those relating to its production system, the system architects can easily identify the parts impacted in the case of a change in requirements. Therefore, it will help them making appropriate decisions to update the product and production system architectures [Mhenni et al. 2018].

Conclusion of the contribution

In this contribution, we have proposed an approach for the joint design of mechatronic products with their production process in order to take into account the interdependence between them. This approach is based on a systems engineering methodology using **SysML** as a modeling language and the **SBCE** method.

By having a single model containing both the product and its production system, all actors in the product design and production process can communicate around this single model. They can define a design space made up of a set of possibilities, and then add design requirements and manufacturing constraints during the development process until a single solution is reached. This allows the design of the product and its production system to be optimized while reducing iterations in the design process. The developed model also allows to easily evaluate the impact of changes occurring during the design of the product or its production system.

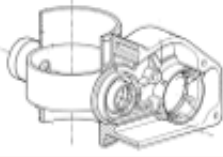



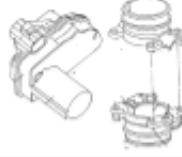













b1.One-bloc, co-molded, bi-plastic	b2.One bloc, molded, plastic	b3.One bloc, molded, metal	b4.Two-blocs, molded, metal	b5.Three-blocs, molded, plastic										
														
G1:Plastic gears molded		G2: Metal gears molded		G3:Plastic gears over molded										
														
R1: Metal regulate air flow system			R2: Metal plate shaft and plastic throttle plate											
														
1S1: One spring			S2: Two-springs											
														
M1: DC Motor			M2: Step Motor											
														
P1: Double track potentiometer			P2: Simple potentiometer											
														
C1: PD Controller			C2: PID Controller											
														
5 Body	×	3 Regulate airflow	×	2 Gears system	×	2 Spring	×	2 Position sensor	×	2 Motors	×	2 Controllers	=	480 potential ETB configuration

Figure 1.21: Possible solutions for each subsystem of the ETB

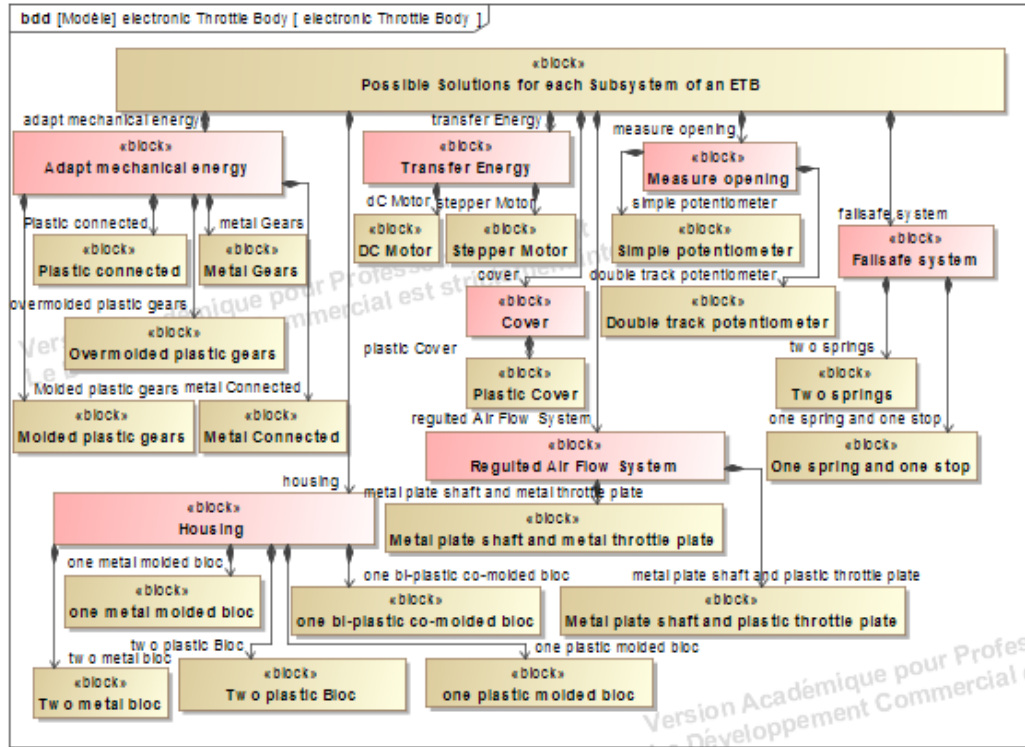


Figure 1.22: Potential ETB configurations represented by a SysML BDD diagram

1.2.1.2 Modeling and evaluation of the reconfigurability of manufacturing systems

This contribution is part of the thesis of Abdelmonaam ABID [Th4] whose work aims at developing a methodology for the analysis of the reconfigurability of manufacturing systems.

Indeed, manufacturing system configuration has a high impact on system productivity. Rigorous verification of manufacturing system reconfigurability through the development of simulation models could reduce infeasible or uneconomic investments. As manufacturing systems become more complex and sophisticated, it is very difficult to easily obtain an executable model from huge system specifications. This contribution presents a methodology and a framework to model and verify a Reconfigurable Manufacturing System (RMS) by simulating configuration alternatives. Figure 1.25 shows the proposed methodological framework for the design of a reconfigurable manufacturing system.

The idea behind this methodology is that multi-agent systems are the best suited paradigm to simulate the reconfigurability of RMSs. In a context of model-based systems engineering with SysML language, where one seeks to centralize the specification and verification while keeping traceability during the development cycle, the issue is that it is difficult to model a multi-agent system of a reconfigurable manufacturing system from SysML specifications. Hence the need to make

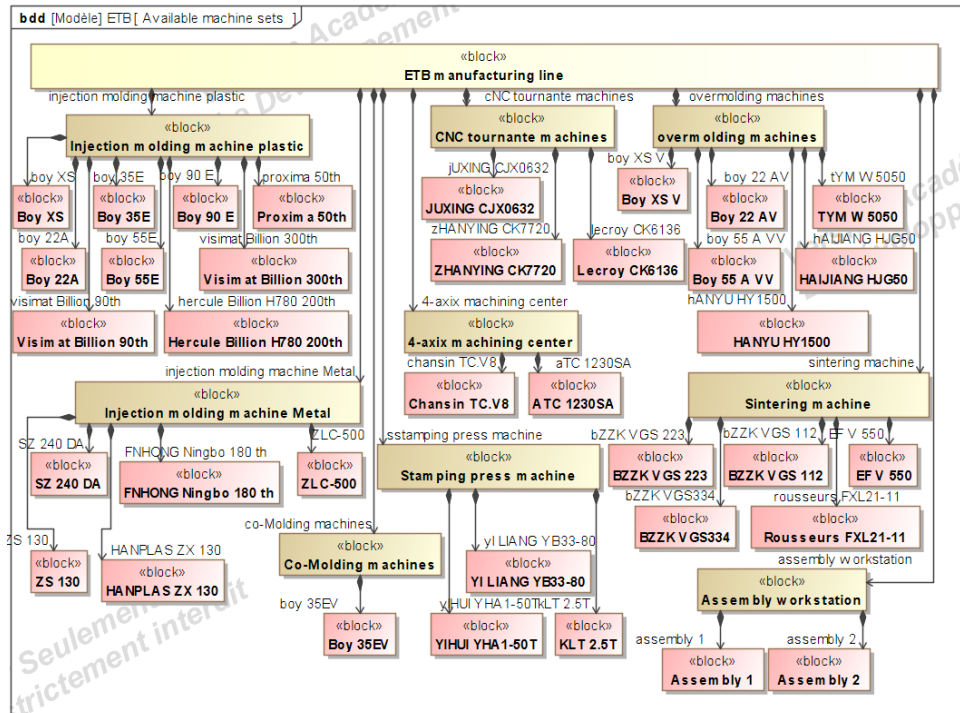


Figure 1.23: The production process and the manufacturing machines of the ETB configurations represented by a SysML diagram

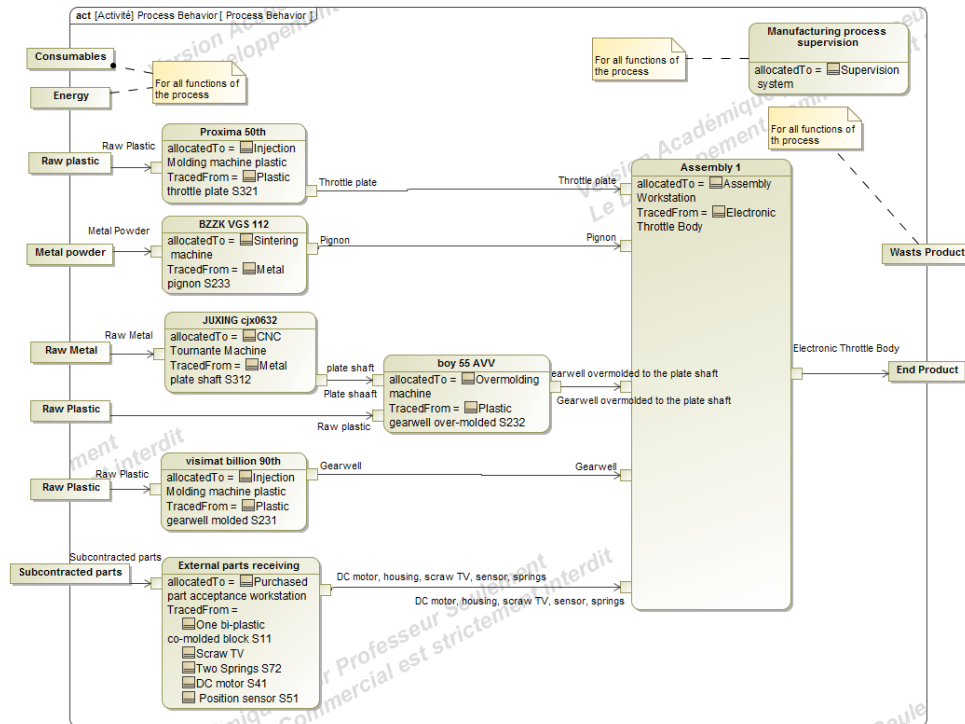


Figure 1.24: The functional structure of the ETB production process

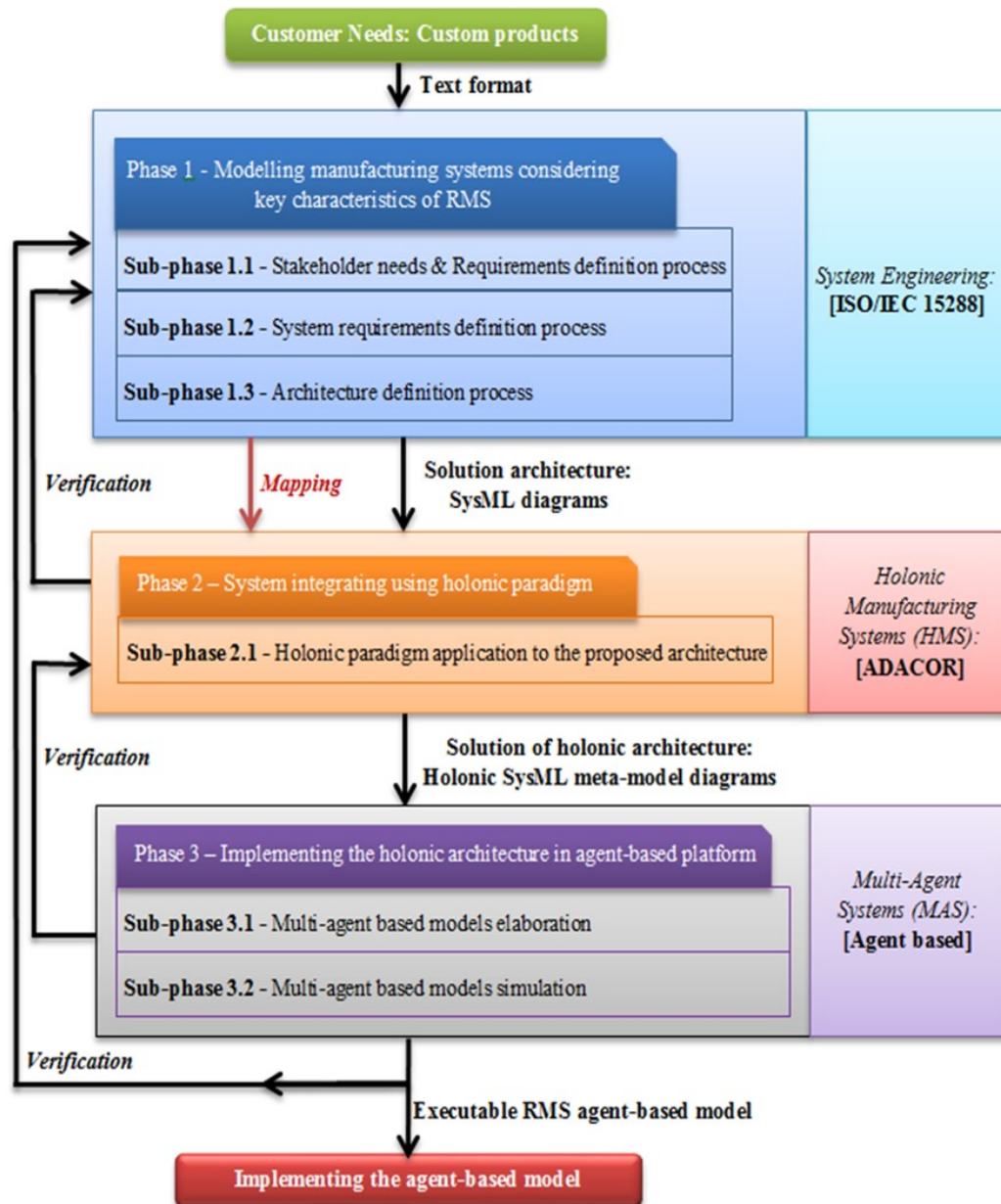


Figure 1.25: Methodological framework for the design of a reconfigurable manufacturing system

transitions to ensure continuity between specification, modeling and simulation. Thus, we propose in this methodology to add a phase of modeling of a holonic architecture from the specifications modeled according to a classical approach of systems engineering (supported for example by the standard ISO/IEC 15288). Then, we detail this holonic architecture to define a multi-agent system that can be easily implemented in an agent-based simulation environment. Indeed, the holon concept and the agent concept are very similar in many aspects such as autonomy, reactivity, rationality, etc. Other details on multi-agent systems will be given in the section 1.4.2 about collaborative design.

The proposed framework, therefore, allows the designer to progress from the conceptual modeling level to the simulation level through a suggested mapping and transformation process. This process consists of 4 phases to be realized within the specification and modeling environment based on the SysML language which are: 1) the specification of the key characteristics of the RMS; 2) Definition of a holonic architecture of the RMS; 3) Detailing the simulation model of the RMS as a multi-agent system; and 4) implementation of the multi-agent system model in an agent-based simulation tool.

In the first phase of the methodology, the idea is to use a systems engineering approach based on SysML, as illustrated in section 1.2.1.1, to specify both the product and the production system. Thus, in this phase we specify the requirements of the product, its behavior, its structure, etc. In the same way, for the production system, we specify the attributes of its structure, behavior, capabilities, etc. This phase is performed in a top-down way to get a set of SysML diagrams that model the product and the production system in a hierarchical manner. The models are established in such a way as to ensure traceability between the product and production requirements, on one hand, and the behavioral and structural modeling elements, on the other.

As mentioned above, the obtained SysML models do not allow a simulation of the production process to analyze the reconfigurability of the RMS. Therefore, in the second phase of the proposed methodology, the obtained system models from the first step are mapped to a holonic architecture. Indeed, Domain Specific Language (DSL) tools implemented with UML-SysML allow us to develop new diagram profiles based on the classical SysML diagrams to customize their use. Therefore we used DSL to develop an extended holonic SysML profile as illustrated in figure 1.26.

The proposed Holonic SysML profile is based on the ADACOR (ADaptive holonic COntrol aRchitecture for distributed manufacturing systems) [Leitão and Restivo, 2006]. ADACOR is one of the architectures suggested in literature to implement manufacturing control systems that require a dynamic and reactive reconfiguration based on distributed and intelligent control approaches. ADACOR merges respectively the benefits of hierarchical and heterarchical control structures using an adaptive mechanism.

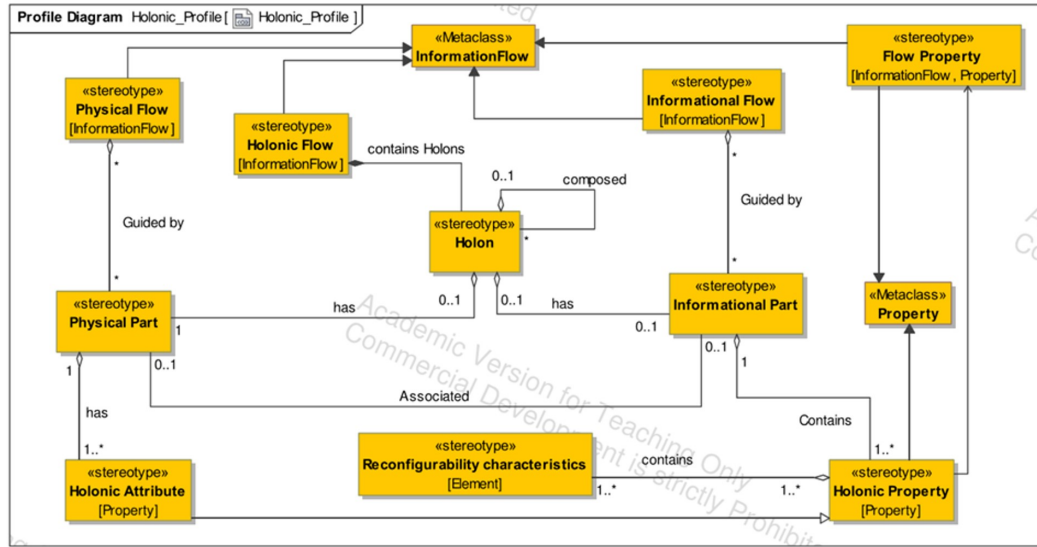


Figure 1.26: Extended Holonic SysML profile

The ADACOR/SysML diagram profile contains attribute stereotypes that allow the RMS designer to specify the following holons:

- The product holon (PH): used to define the necessary knowledge for the general operation of the manufacturing system and for selecting the general strategies that respect the planned objectives. The product holons represent the products of the RMS.
- The task holon (TH): used to specify the knowledge to manage the execution of the strategies to attend the market demands.
- The supervisor holon (SH) : used to describe all the knowledge to coordinate the holons of lower hierarchical levels, such as the coordination of the holon's operations.
- The operational holon (OH): used to represent the RMS physical resources that have specific control devices for its automatic operation. It is also used to model the behavior of these resources in accordance with their objectives and capabilities.

The third phase consists in the definition of a metamodel of the multi-agent system using SysML. Indeed, the multi-agent system has the same architecture as the holonic architecture. However, the multi-agent metamodel has more attributes and details that are required for simulation purpose. Thus, we specify the parameters of each agent, the variables (which will vary during the simulation), the operations that represent the functions, etc.

The fourth and last step of the methodology is the implementation of the multi-agent simulation model in an agent-based simulation environment. The implementation is dependent on the details specified in the multi-agent system metamodel and the software tool supporting the multi-agent simulation.

To validate this approach, a case-study of an industrial manufacturing system is considered to simulate its reconfigurability for the purpose of analyzing its productivity. The considered case-study is based on a steel converter process as illustrated in figure 1.27. The final product of this process is simple steel bars, but as we will see in this example the production system is not so simple to model and simulate. For this reason, we preferred to validate the approach on a simple mechanical product and not a complex mechatronic product.

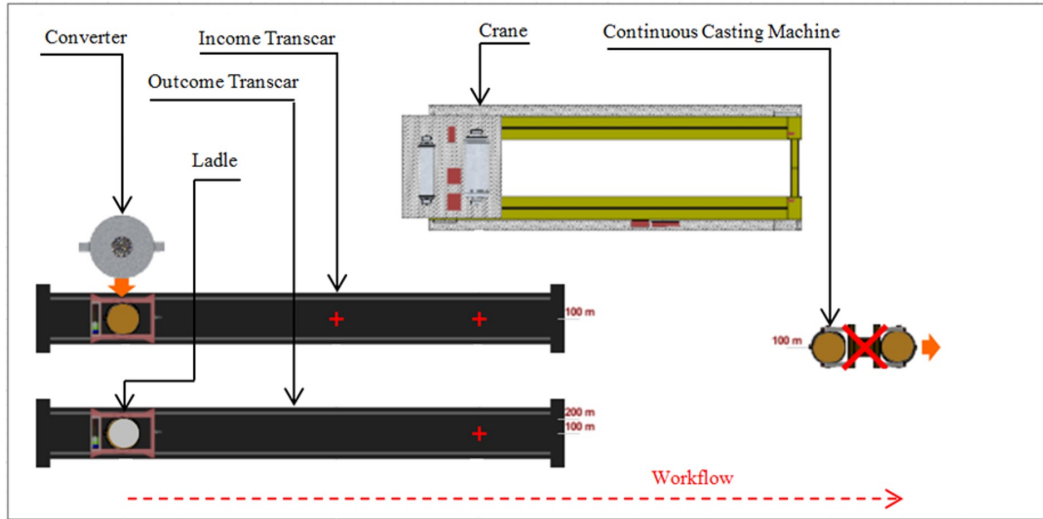


Figure 1.27: Case study: Steel converter manufacturing system

Based on the planned orders, the steel converter process starts with the transport of molten iron from the blast furnace to the converter. The molten iron is added to the scrap steel, which lowers its temperature. The converters will be loaded with up to 250 tons and will turn it into steel in about 20 minutes. Then the converter is tipped and the steel is poured into a ladle.

During the process, a steel ladle is fed to the continuous casting machine (CCM) by an overhead crane which is transported by the income transcar. In order to make the casting sequence possible, the whole unit is lifted by the crane onto a rotating turret. Then, in order to allow the steel to flow inwards from the ladle, the ladle nozzle is opened. The final task is to horizontally exit a custom solid steel slab from the discharge end of the machine where it is automatically cut to the desired lengths. Simultaneously with the casting operation, the empty ladle is loaded by the crane and discharged into the outcome transcar.

The main challenge of this case study is to develop a simulation model for the productivity analysis of the manufacturing system considering different configura-

tions: number of cranes, number of continuous casting machines, etc. The casting quality can be improved in this case by reducing the discontinuity times. Thus, productivity is measured by simulating the usage rate of the machines (CCM, Crane, etc.).

The application of the proposed methodology starts with the elaboration of different SysML diagrams showing the different points of view of the production system. Thus, the figure 1.28 shows the block definition diagram (BDD) describing the main parts of the production system.

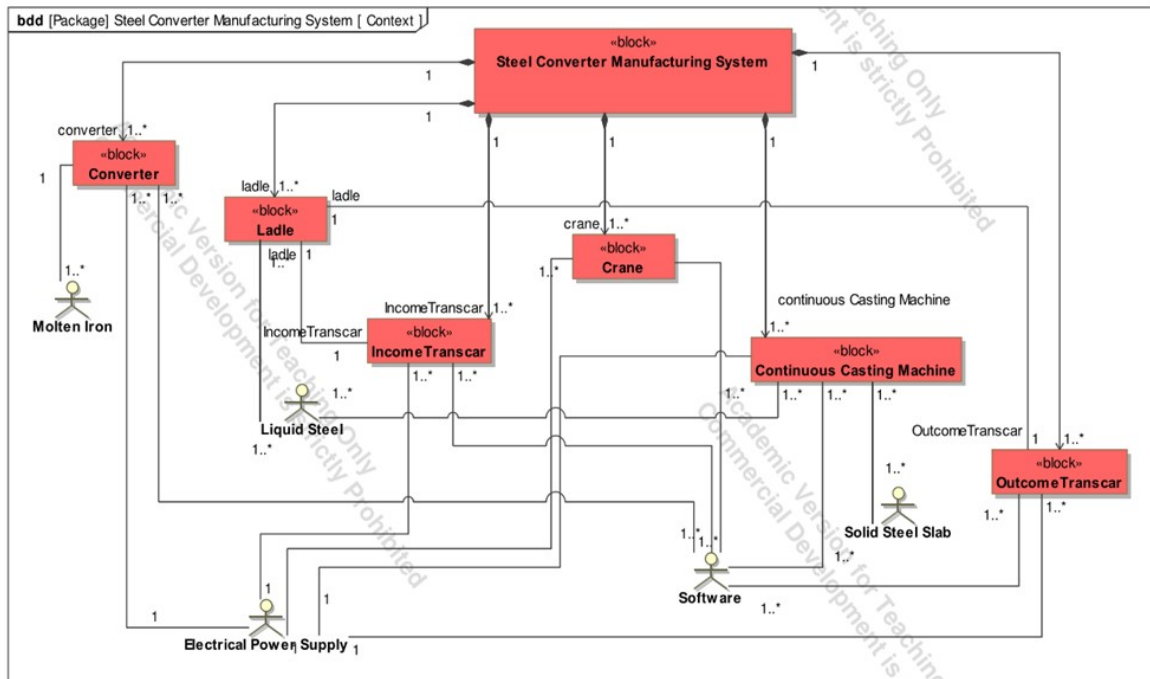


Figure 1.28: Phase 1: SysML model of the steel converter system

To detail the structure of some physical parts of the production system other SysML diagrams can be added. For instance, the continuous casting machine is detailed with an internal block diagram (IBD) as illustrated in figure 1.29.

The traceability between the physical elements of the production system and the requirements could be ensured with a requirements diagram that can link structural elements with requirements via "satisfy" relationships. The traceability relations between the requirements themselves could be of the "derive requirement" type as illustrated in the figure 1.30.

The final simulation of the production system requires the modeling of the behaviors of the different parts forming the system. Thus, the behavior of the continuous casting machine has been modeled with a state-machine as shown in figure 1.31.

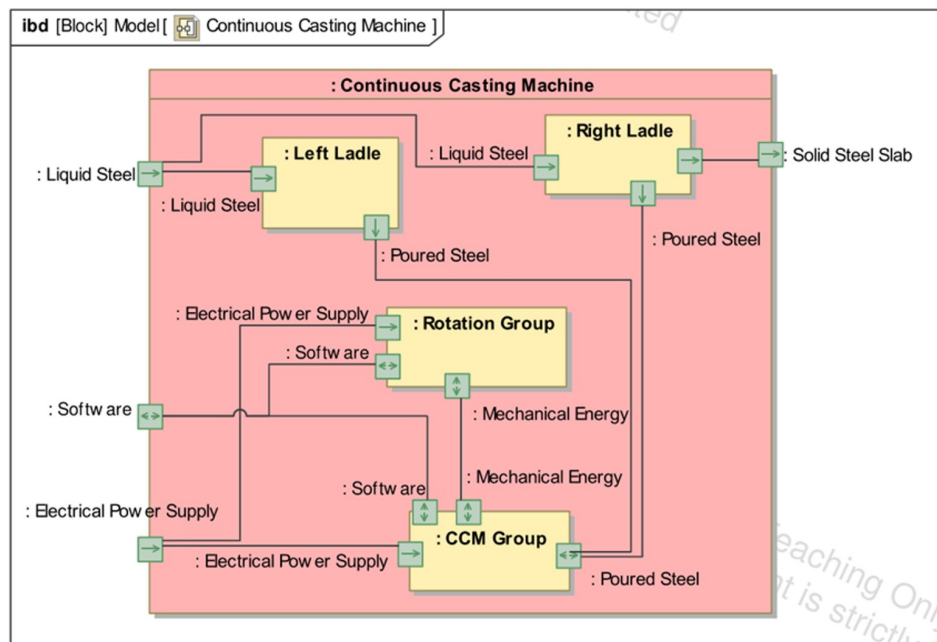


Figure 1.29: Phase 1: Internal structure of the continuous casting machine

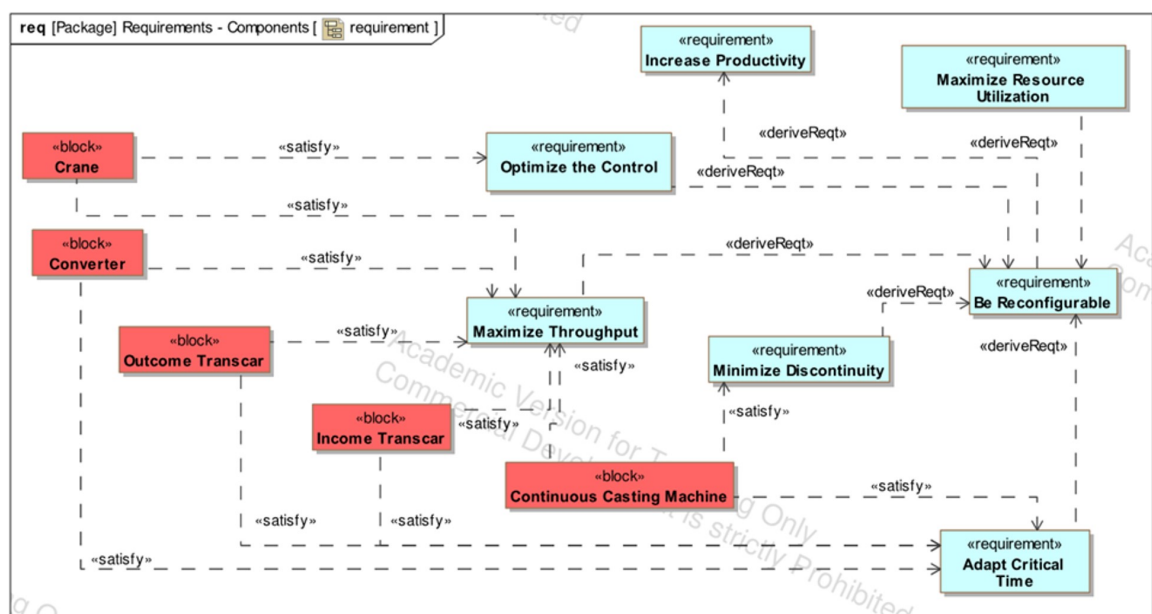


Figure 1.30: Phase 1: Traceability between requirements and physical elements of the production system

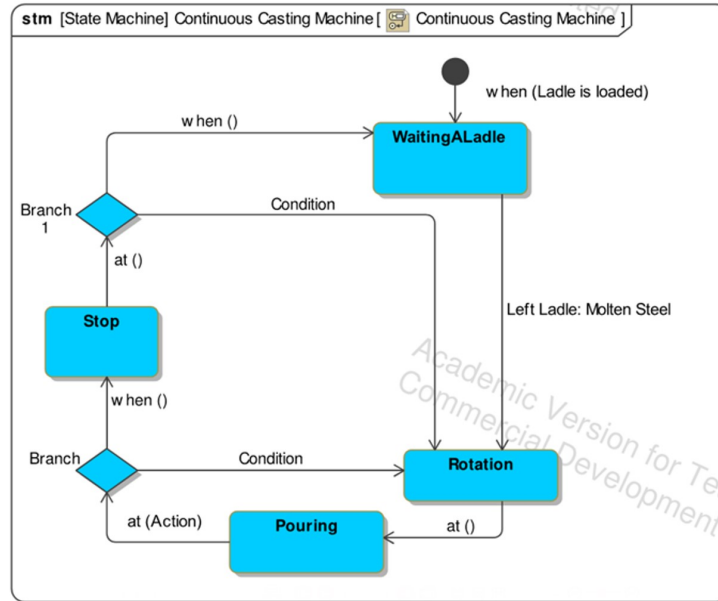


Figure 1.31: Phase 1: A state-machine modeling the behavior of the continuous casting machine

The second phase of the proposed methodology consists in defining the holonic architecture of the production system using the holonic profile defined previously. Figure 1.32 shows the elaborated holonic architecture.

The third phase of the proposed methodology is to define a multi-agent system of the production system with SysML. Figure 1.33 shows the SysML metamodel of the multi-agent system model to be implemented for the reconfigurability simulation.

The SysML metamodel contains details about the agents. For instance, the crane agent contains details such as crane position for 3D animation, maximum horizontal speed, maximum weight, etc. We can also find the list of operations associated to each agent. For the case of the crane agent, operations like make first move, set task parameters, wait task, etc. are declared.

These operations are implemented using the AnyLogic software, which supports multi-agent simulation. The figure 1.34 shows the simulation results for the steel converter process with two cranes and three continuous casting machines.

The verification process is based on the productivity ratio, throughput rate, machine utilization rate and discontinuity time. In this way, the operation discontinuity rate of each physical resource is verified. The simulated configurations were with 1, 2 and 3 cranes combined with 1, 2 and 3 continuous casting machines. The best configuration obtained by simulation is the solution with 2 cranes and 3 continuous casting machines, which reduces the discontinuity time

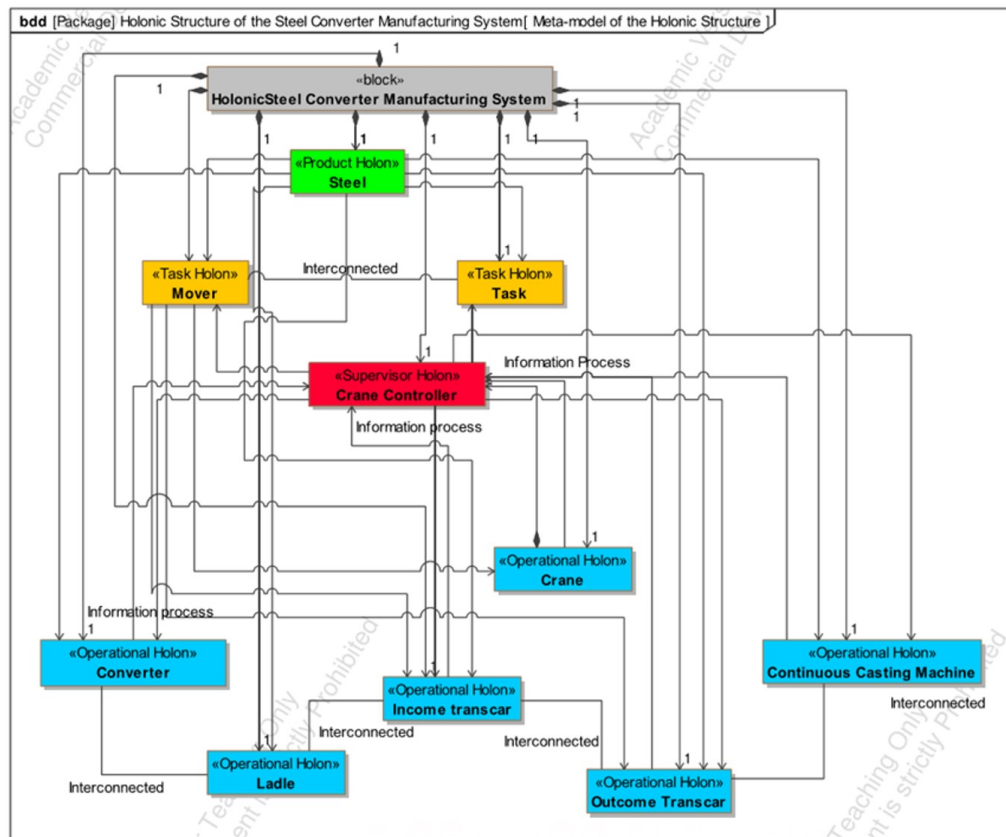


Figure 1.32: Phase 2: Holonic architecture of the production system

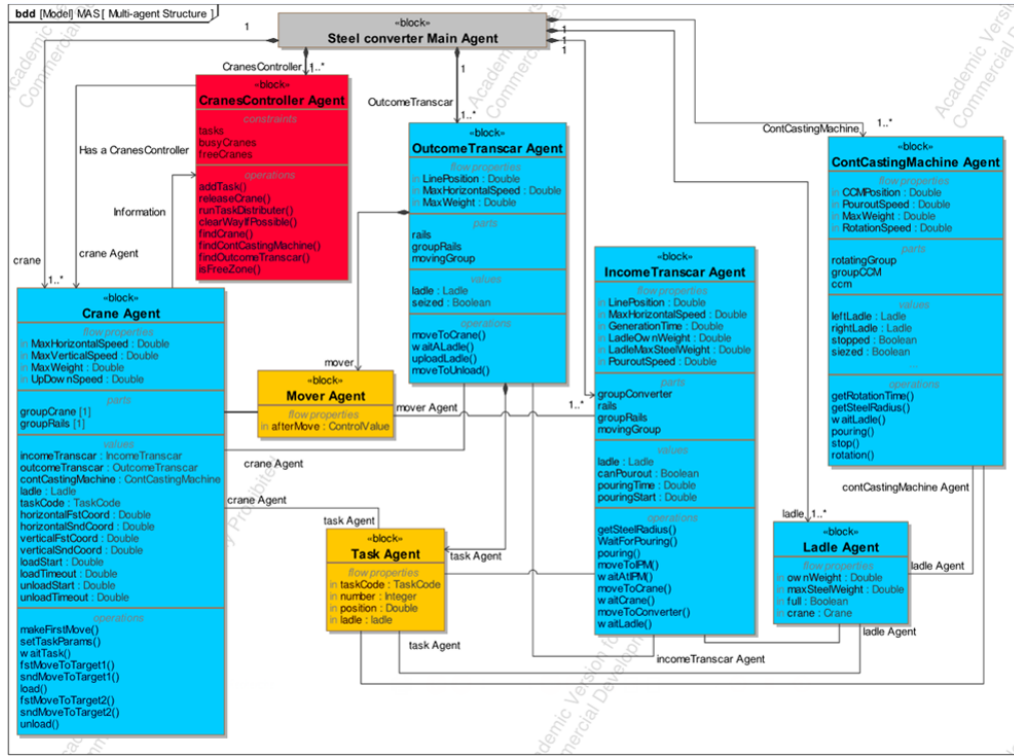


Figure 1.33: Phase 3: Multi-agent architecture of the steel converter system

to 10.3 minutes out of a simulated working time of one hour. The productivity ratio for this configuration is 80.7%, which is also better than the other configurations. Other results from the simulation include CCM and crane utilization rates.

To conclude this contribution, the design issue for this work was how to simulate reconfigurability of manufacturing systems under the constraint of requirement modifications during the design process. For this, a SysML-based methodology was developed in order to reduce the gap between the specification level supported by MBSE and the verification phase supported by multi-agent simulation, while ensuring traceability between requirements and simulation results. This methodology is based on the adaptation of a systems engineering approach using SysML by adding a phase to describe the holonic architecture of the production system and then adding the specification details of a multi-agent simulation model. The last phase is to implement the simulation model in a multi-agent simulation tool. Finally the approach has been validated with a case study of steel converter process.

1.2.2 MBSE approaches to integrate DFX for mechatronics design

DFX allows the integration of production constraints in the upstream phases of mechatronics design. In a MBSE-based design methodology, this requires modeling of DFX metrics and linking their components with the Key Value Attributes (KVA)s

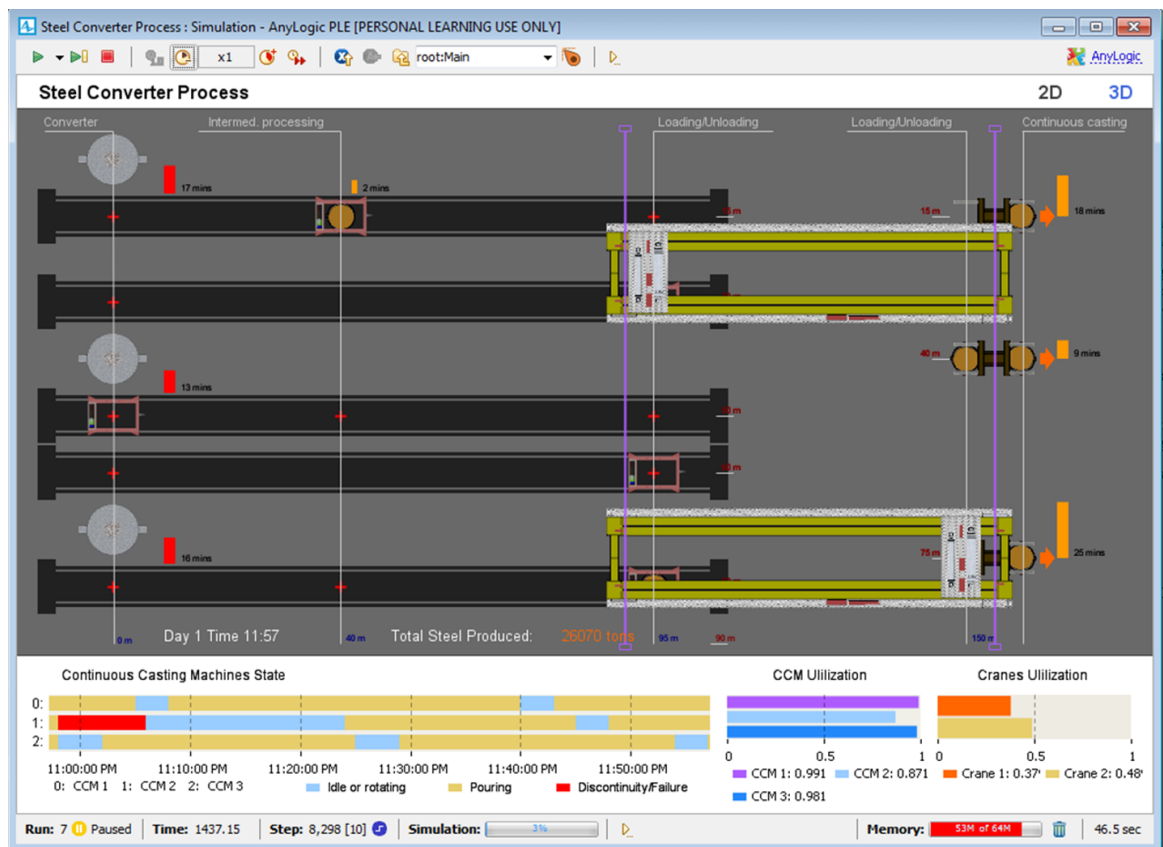


Figure 1.34: Phase 3: Multi-agent simulation model implemented using AnyLogic software

specified with the MBSE. Thus, in this section we will detail the following two contributions:

- a MBSE approach for integrating manufacturability and reliability in the design process of mechatronic systems.
- a contribution to the integration of assemblability in a MBSE approach.

1.2.2.1 MBSE approach for integrating manufacturability and reliability in the design process of mechatronic systems

This contribution is part of the thesis of Mohamed Firas BORCHANI [Th5], which objective is to develop an approach allowing to reduce the number of possible solutions of a mechatronic product based on the two criteria reliability and manufacturability in the preliminary phases of design.

A review of works incorporating manufacturability and reliability in the literature showed several shortcomings:

- Use of several types of standards for data exchange between engineers and experts, which delays product development time.
- Developing a manufacturability analysis system or software is expensive and requires complex coding to extract manufacturing entities and compare them to geometric rules. In addition, when dealing with a complex shaped part, the system may misinterpret the design information.
- The product development framework does not allow stakeholders to exchange ideas and areas of analysis to optimize the part design.
- The manufacturability analysis is often based only on a single manufacturing process, which in most cases is machining. Other processes may offer good value for cost.
- Constraints related to the manufacturing resources available at the production sites are not taken into account in the manufacturability study.

To overcome these shortcomings, the objective of this contribution is to develop an approach integrating the evaluation of manufacturability and reliability in the design process based on the SBCE principles and using SysML language.

figure 1.35 illustrates the idea of the approach.

Compared to a classical systems engineering approach using MBSE tools, the approach proposes to use a new SysML profile (figure 1.36) allowing experts from different domains (manufacturing, reliability, etc.) to bring their contributions at different design levels according to the SBCE principles. In this diagram we define the new KVA associated to DFX metrics: manufacturability, reliability, durability, cost, etc. For instance reliability attributes are associated to the

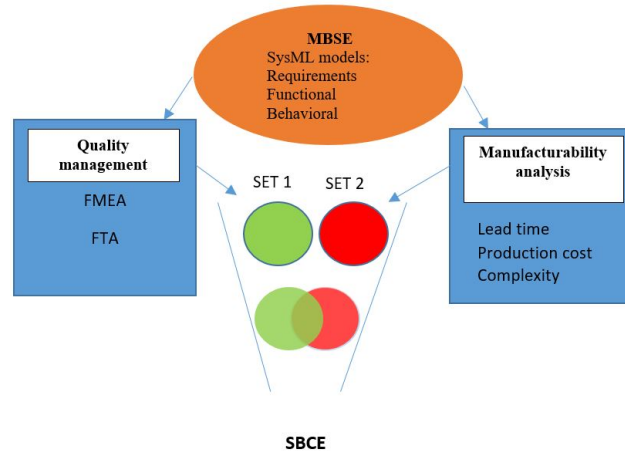


Figure 1.35: Interdisciplinary collaborative mechatronics design using MBSE and SBCE

system/sub-system activities that could represent the system functionalities. On the other hand, manufacturability attributes are associated to both system activity and process features, which could represent a manufacturing operation.

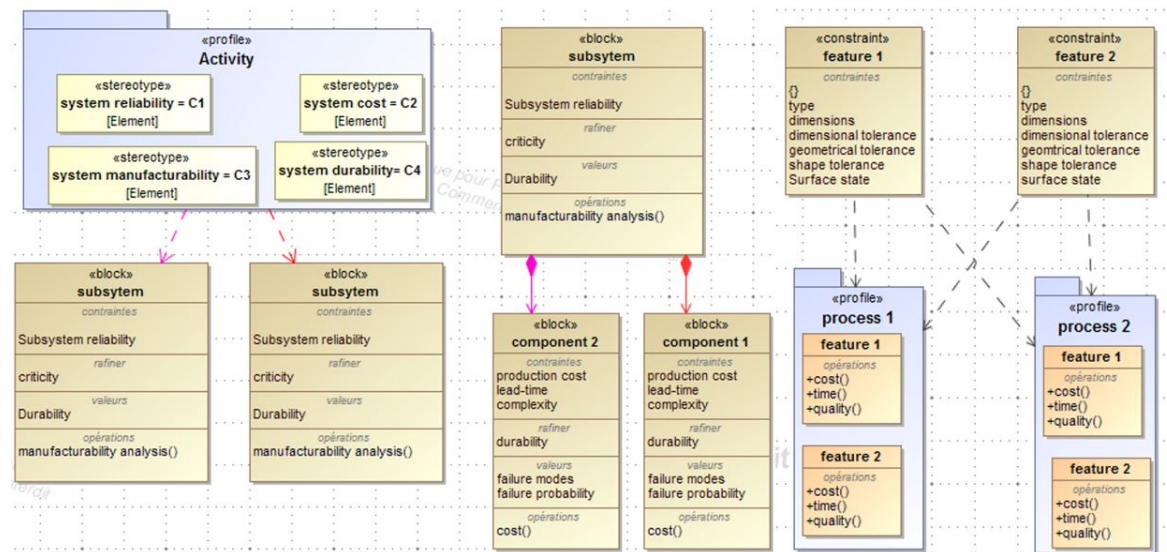


Figure 1.36: New SysML diagram profile for manufacturability and reliability analysis

Since SysML tools allow to define the allocations between system activities and process operations, on the one hand, and design requirements on the other, this will facilitate the mapping between the new attributes related to DFX constraints and the design requirements.

To create the solution set, we use the allocation possibilities of SysML (allocation

matrices) to associate a set of physical solutions to a system functionality and a set of realization processes for the physical components.

The design process starts by associating values to the **KVAs** components either from the capitalized experience of the company, or by consulting experts or simply by looking in the catalogs of the suppliers to seek information, such as failure probability for reliability assessment. Next, the **KVAs** are sorted using an analytical hierarchy process (AHP), where the sorting criteria are set in advance according to the company's needs and the recommendations of the various stakeholders.

To map the design space, design participants and subsystem teams define the scope of the design work required as well as the feasible design options/regions.

The design solution set is exported in the form of a graph (GraphML format) for the evaluation of the **DFX** metrics and the reduction process of the number of solutions.

The evaluation of **DFX** metrics is done using mathematical models and algorithms to calculate the values of the metrics from the **KVAs**. Specifically, for the reliability assessment, a Failure Modes and Effects Analysis (**FMEA**) method was used in this work. The idea is to outline Severity (S), Occurrence (O) and Detection (D) rating scales. Severity indicates the gravity of the effects of a failure mode, occurrence denotes the probability of a failure occurring, and detection measures a failure's visibility that is the attitude of a failure mode to be identified by controls or inspections. Each parameter is correlated to a score range (chosen to be from minimum of 1 to a maximum of 5 in this work). To calculate and compare subsystem's criticities, each component criticity is calculated using risk priority number (RPN), which is defined as the product of the three components (parameters) S, O and D:

$$RPN = S \times O \times D \quad (1.1)$$

Once the reliability metric is evaluated for each solution in the solution set, a developed algorithm is used to reduce the solution set by eliminating weak solutions and those that do not meet the design criteria. The flowchart describing the reliability assessment process is presented in figure 1.37

Thus, for each **DFX** metric, mathematical models and a solution reduction procedure need to be established.

To validate this approach, an electric compressor system was considered as a mechatronic product used for air conditioning application in automotive. Figure 1.38 shows a picture of the considered system and its components.

Electric compressors have the advantages of low refrigerant emissions, flexible packaging, and efficient variable-speed operation. This automotive module constitutes the interface between the low and high-pressure sides of the refrigerant circuit.

The design space in this case study is limited to 15 solutions obtained by combining 3 possibilities of motors (Brushless, step by step and DC motor) with 5 possibilities of compressing techniques (Scroll, axial, lobe, screw and centrifugal). For the other compressor components, we have made the assumption that there is one unique

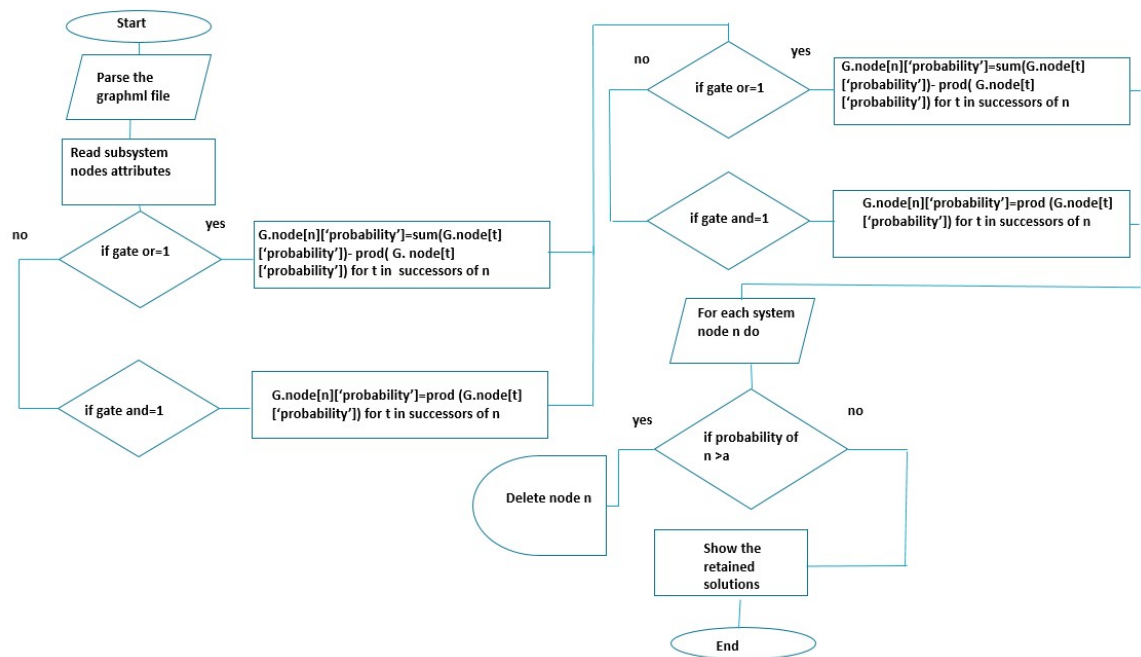


Figure 1.37: Reliability assessment flowchart

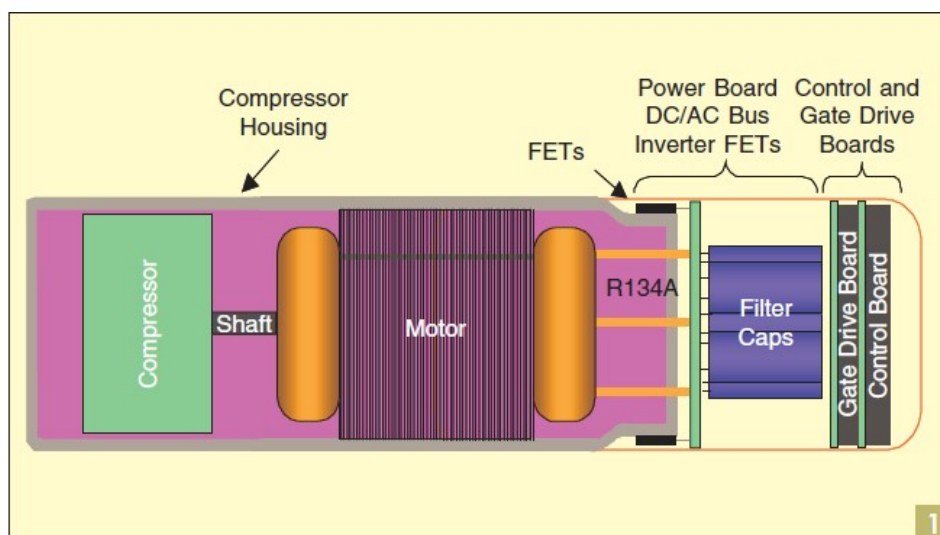


Figure 1.38: Case study of an electric compressor for automotive application

solution for illustrative purposes.

After applying the proposed methodology only three final solutions were retained. The final decision could be made with a trade-off by comparing the three solutions using, for example, a radar chart as illustrated in figure 1.39

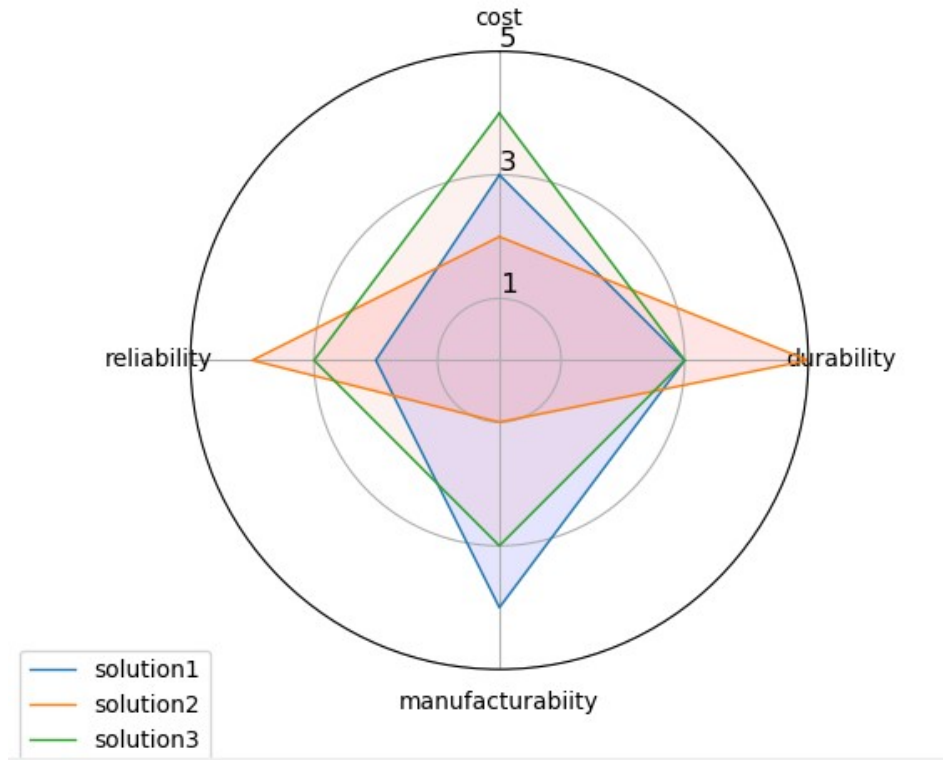


Figure 1.39: The radar chart for the three final retained solutions

To conclude this contribution, a new SysML profile has been developed to integrate the attributes related to the DFX metrics, especially for manufacturability and reliability. This will allow the different actors of the mechatronic product development cycle to add their knowledge about the product in the preliminary design phases. For each metric, mathematical models and logic rules for evaluation are needed, and algorithms to reduce the design space must also be developed. The final choice of the solution could be made based on a trade-off analysis comparing the different DFX metrics with the system performance metrics.

1.2.2.2 Contribution to the integration of assemblability in the MBSE approach

This work is part of the thesis of the doctoral student Rihab Brahmi [Th7]. We position ourselves in the context of the development of a complex mechanical or mechatronic product where assembly plays an important role in the design. In this

situation, a collaboration between the system engineer and the 3D CAD expert is necessary to analyze the impact of assemblability on functionality. Indeed, before embarking on the manufacture of such complex products, it is necessary to perform a functional verification of the assembly, which determines whether individual components are required and verify their functions. Functional analysis also allows the verification that all product functions are achieved by the design and that all the design requirements are met. The analysis of the assemblability, on its side, allows the systems engineer to determine:

- whether the number of components can be reduced,
- whether some components can be replaced by others allowing self-positioning and self-inserting,
- the ease of assembly of parts with clearance
- if the need for reorientation during assembly can be minimized
- if parts that cannot be installed incorrectly can be eliminated
- if symmetrical parts can be maximized
- etc.

In this contribution, we propose an approach aiming at improving the collaboration between the systems engineer and the CAD expert.

The proposed approach is based on three phases which are :

- Phase 1 - a preliminary MBSE study: in this step, a global study of the product from a functional point of view is performed by the systems engineer, with a global specification of the assembly requirements.
- Phase 2 - a 3D CAD design development: in this step, the designer proposes a 3D design that could meet the need, with a the preliminary results of the assembly analysis.
- Phase 3 - enrichment of the MBSE and validation of the design: in this step, the systems engineer checks if the proposed design solution is satisfactory by verifying the compatibility between the system functionality and its assemblability.

The proposed methodology is illustrated directly on a case study of a bicycle pedal with flashing LED lights. The product need can be described as following: to reassure parents who want their children to remain visible, even if their dynamo lamp does not work, we need to develop a pedal that increases the cyclist's visibility and requires no maintenance: no batteries, no wires, no moving parts and no risk of failure. The illustrative example is inspired from an industrial example designed



Figure 1.40: Bicycle pedal with flashing LED lights

and manufactured by the company Pedalite, which is a luminous pedal named KPL200 as presented in figure 1.40.

This product will be considered as a baseline for the development of a simpler, more efficient and innovative mechatronic product. The functional specifications can be presented as following: when pedaling, a small generator integrated to each pedal produces current and powers the flashing lights. The surplus energy produced will be stored and then released when the cyclist is not pedaling (downhill, at a standstill, etc.). The battery life should be at least 5 minutes.

The first step of the first phase (preliminary MBSE study) consists in deciphering all the needs. For instance, the primary need of the pedal is to improve road safety by increasing the visibility of cyclists at night, even in adverse weather conditions.

The second step of the first phase is the definition of the functional requirements. The functional description must be complete, coherent and have a very clear structure in order to be understandable during the validation phase. MBSE with SysML were chosen for their potential to describe the product from several points of view. For the example discussed, a functional requirements diagram is developed as illustrated in figure 1.41 to present a part of the requirements related to the transmission of mechanical energy.

After studying the product from a functional point of view, the systems engineer can propose an initial structure of the assembly. Modeled with SysML block definition diagram "BDD", this structure essentially contains the parts that are the basis of the system's operation. As illustrated in figure 1.42, the initial structure of the example treated can be composed of five essential parts which are: a pedal body and a pedal axle used to transmit the mechanical energy provided by the cyclist during pedaling; a double intermediate wheel that multiplies the rotation of the axle; a generator to convert the kinetic energy into electrical energy to have light flashes and an integrated circuit used to store the surplus of energy produced.

The last step of the first phase is to define the assembly requirements and their traceability with other design specification (requirements, functional, structure and behavior). This step has a major importance in improving the quality of the assembly, because it allows the system engineer to quickly identify the impact of any

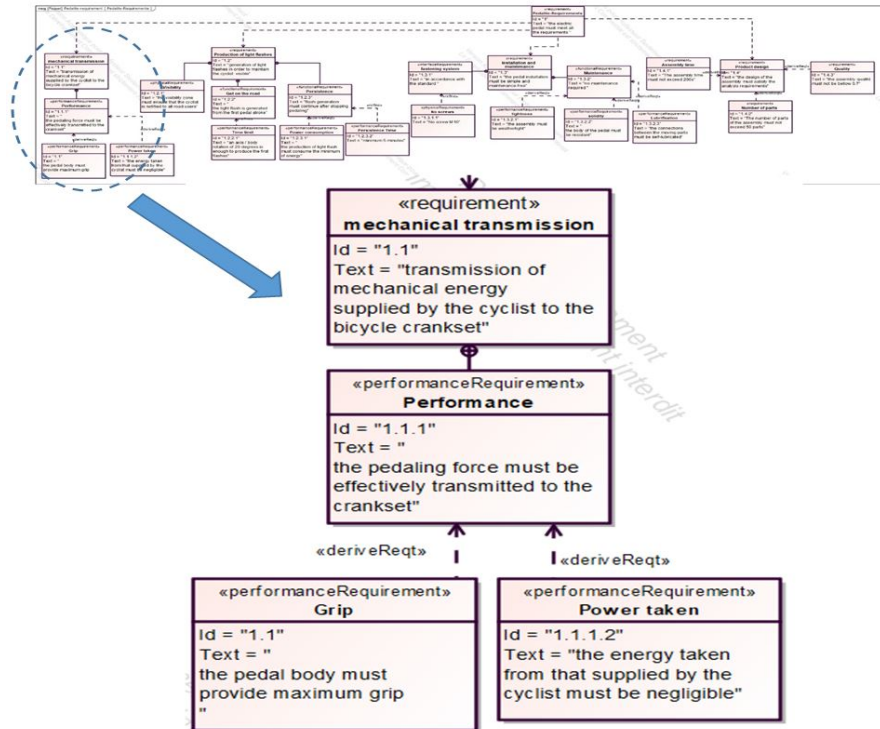


Figure 1.41: Pedal requirements diagram

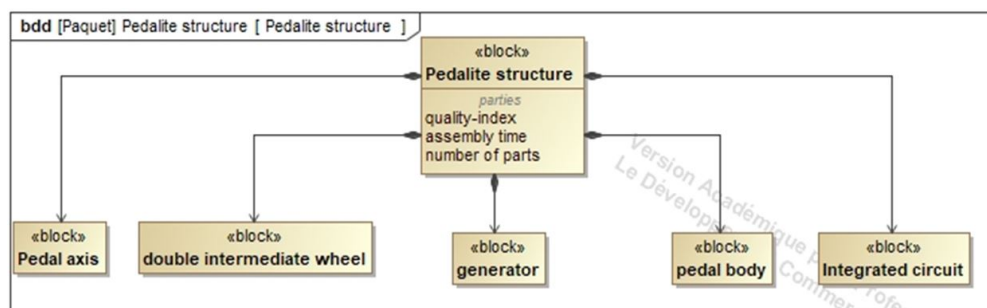


Figure 1.42: Initial structure of the pedal system

change in one of the requirements on the parts in the assembly. For the case-study considered, figure 1.43 shows the traceability links between some requirements and related parts such as pedal body and the axis of the pedal.

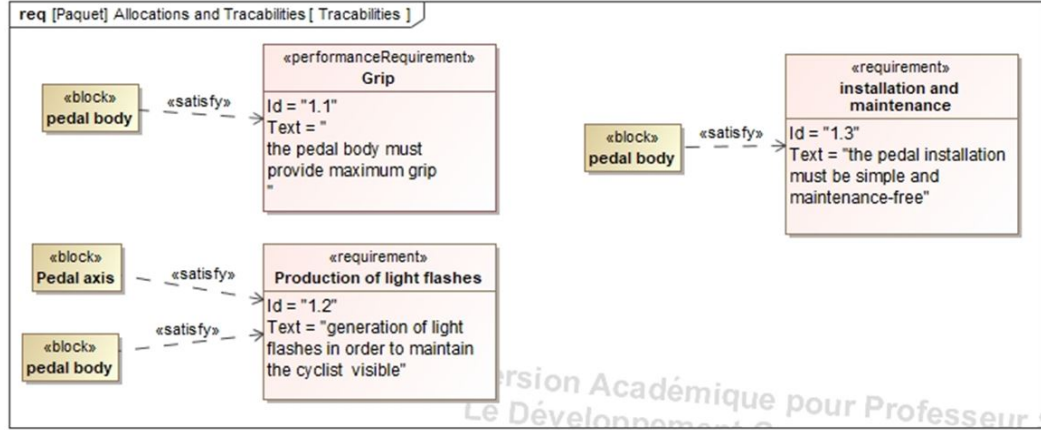


Figure 1.43: Allocation and traceability of the pedal

In the second phase of the approach, a 3D CAD design should be developed or adapted from a previous work while trying to satisfy the requirements specified in the first phase. For our illustrative example, figure 1.44 shows a representation of the 3D design which is composed of 29 parts.

Table 1.1 illustrates the nomenclature of the proposed solution.

Rep	item	Nb	Rep	Item	Nb 1
1	Pedal body	1	12	Screw	2
2	Pedal axle	1	13	Contact	1
3	Sealing ring	1	14	Electronic card	1
4	Bearing	2	15	Mechanical belt	2
5	Spacer	1	16	Insert	1
6	Reflector cover	2	17	Inner cover	1
7	Reflector	2	18	Outer cover	1
8	CHC screw	2	19	Engine body	1
9	Generator	1	20	Motor wheel	2
10	Connector	1	21	Motor wheel 2	1
11	Bearing bracket	1	22	Motor cover	1

Table 1.1: Nomenclature of the pedal

The proposed design solution must be evaluated from an assembly point of view. For this, we propose to investigate three KVAs, which are: the maximum number of parts, total assembly time (TAT) and assembly efficiency (AE).

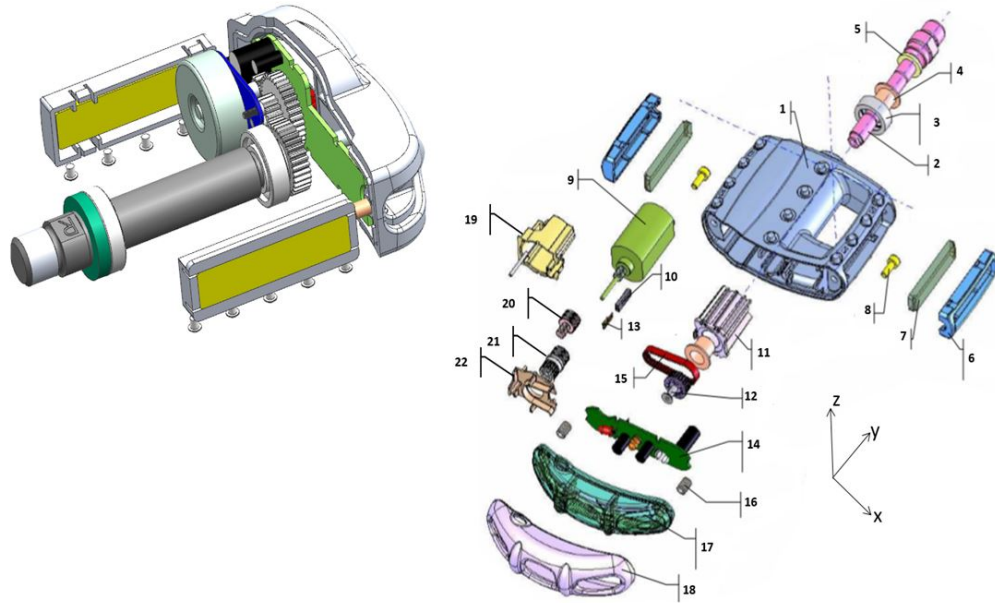


Figure 1.44: CAD and exploded views of the pedal bike design

With the Maximum number of parts KVA , the 3D CAD designer analyzes the assembly by checking whether the total number of parts is optimal or can be reduced. The proposed KVA is based on a procedure that attempts to minimize the total number of parts. According to Boothroyd [Boothroyd and Alting, 1992], the optimal number of parts in the assembly design can be calculated using a procedure based on the answer to four main questions:

1. Is the part a base?
2. Is there a functional relative movement between the part and the others already assembled?
3. Does the part need to be made of different materials or be isolated from other parts already assembled?
4. Does the part need to be separated from other parts to allow for its assembly or disassembly?

For each part of the assembly, if the answer to all four questions is "no", the part in question is not needed and can be eliminated from the assembly design. This will determine the minimum number N_{min} of parts needed for assembly.

The second step of the assemblability analysis is to evaluate the possible assembly sequence plans (ASPs). Figure 1.45 shows a list of possible ASPs.

The evaluation of the ASPs is performed by determining the total assembly time TAT and the assembly efficiency (AE).

ASP1	(2)	(5)	(4)	(3)	(1)	(11)	(9)	(19)	(10)	(13)	(20)	(12)	(21)	(15)	(22)	(14)	(17)	(16)	(18)	(7)	(6)	(8)
ASP2	(1)	(7)	(6)	(8)	(2)	(5)	(4)	(3)	(11)	(9)	(19)	(10)	(13)	(20)	(12)	(21)	(15)	(22)	(14)	(17)	(16)	(18)
ASP3	(1)	(11)	(9)	(19)	(10)	(13)	(20)	(21)	(12)	(15)	(22)	(7)	(6)	(8)	(2)	(5)	(4)	(3)	(14)	(17)	(16)	(18)
ASP4	(1)	(9)	(19)	(10)	(13)	(11)	(20)	(12)	(21)	(15)	(22)	(14)	(17)	(16)	(18)	(7)	(6)	(8)	(2)	(5)	(4)	(3)
ASP5	(2)	(5)	(4)	(3)	(1)	(7)	(6)	(8)	(11)	(9)	(19)	(10)	(13)	(20)	(12)	(21)	(15)	(22)	(14)	(17)	(16)	(18)

Figure 1.45: List of possible ASPs

For the total assembly time TAT , according to Boothroyd, experimental tables can be used to calculate the handling and the insertion time of each part during the assembly process. The assembly efficiency (AE) KVA can be calculated using the following formula:

$$AE(\%) = \frac{T_{min} \times N_{min}}{TAT} \times 100 \quad (1.2)$$

Where:

- T_{min} : Minimum assembly time in the industry's assembly line.
- N_{min} : Minimum theoretical number of parts.
- TAT : Total Assembly Time of the product.

The current work of this thesis focuses on developing algorithms to automate the reduction of assembly sequence plans and to enrich the SysML model with new assembly data and with the results of the assembly analysis. For the example discussed, the updated structure and a zoom on the pedal axle part are shown in figure 1.46.

The last phase on the proposed approach is to validate the design assembly from a functional and assembly point of views. This work is in progress for the development of algorithms to evaluate a set of assembly sequences, by taking into account the functionality and performance of the system, and automatically reduce the number of solutions to keep only the most efficient ones.

To conclude this contribution, the objective is to validate the 3D design solutions proposed to meet a specification prescribing the functionality of a mechanical or mechatronic product. The evaluation criteria considered here are the functionality of the system and its assemblability. This work is in progress with a thesis defense planned for 2022. The outcome of this work could be coupled with the two previous contributions to propose a DFX platform for mechatronic products.

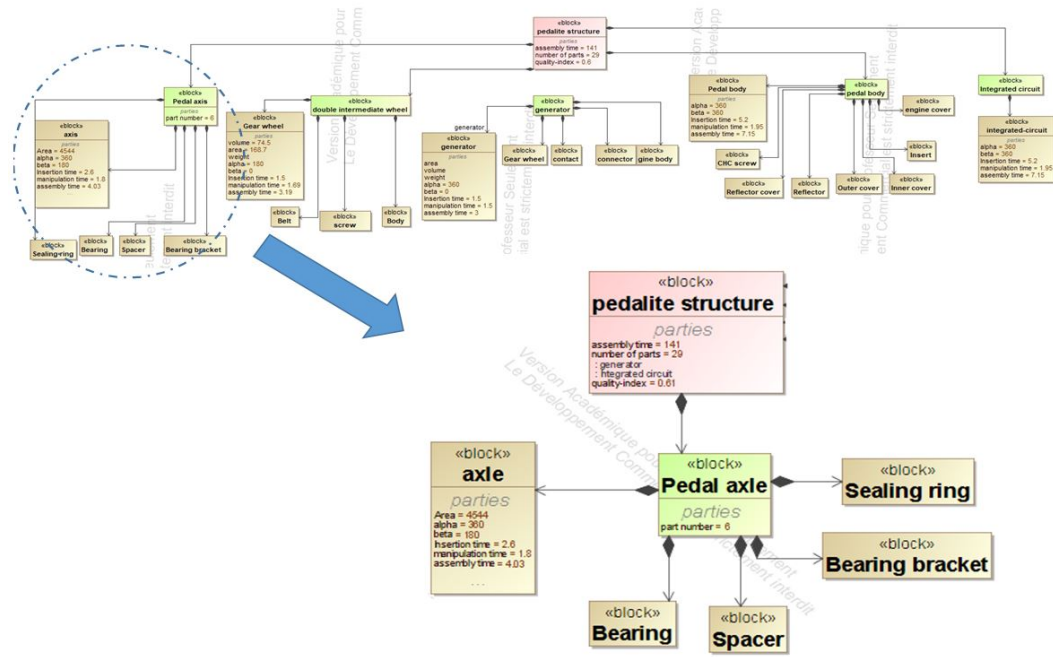


Figure 1.46: Enriched structure of the pedal in the MBSE environment

1.3 Multiphysics simulation methods for mechatronics design

1.3.1 Contribution to the development of compact analytical models for vibration analysis of mechatronic systems

Mechatronic modeling tools (such as Modelica/Dymola) integrate multidisciplinary modeling libraries allowing the simulation of multi-physics phenomena while ignoring the details of the geometry. However, for a good choice of the mechatronic system architecture solution in the preliminary design phases, some physics require a level of detail requiring the consideration of a certain dimension of the geometry. This is the typical case of the vibration of flexible structures which could have an impact on the decision of choosing the mechatronic design solution. This contribution is part of the work of Ghazoi HAMZA [Th1], whose objective is to develop compact models to model the vibration to simulate the vibration in the upstream phases of mechatronic design.

The main contribution of this work is to propose a method for the preliminary design of mechatronic systems based on an analytical approach, taking into account the effect of vibrations. For this, compact analytical models have been developed and implemented in Modelica/Dymola to study the vibration behavior of mechatronic systems. The idea is to develop libraries of Modelica components for multiphysics modeling that are stable and efficient for the preliminary design

of a mechatronic system. This approach should provide the designer with a large space of solutions to choose from. We illustrate this approach with a case-study of a mechatronic system which is a wind turbine.

1.3.1.1 Analytical model of a flexible beam with a rigid mass

The presented model focuses on the deflection of the system shown in Figure 1.47.

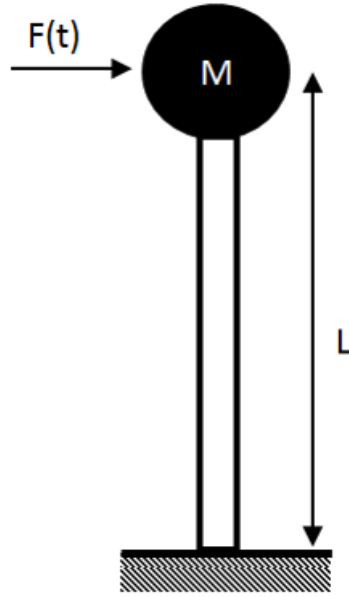


Figure 1.47: Flexible beam with a rigid mass

The system consists of a flexible cantilever beam supporting a rigid mass at its free end. The Euler Bernoulli beam theory is used to predict the deflections. Secondary effects such as shear, axial deformations and rotational inertia are neglected. The equation of motion governing the transverse vibration of the beam can be formulated as follows [Bambill et al., 2003]:

$$EI \frac{\partial^4 w(x, t)}{\partial x^4} + \rho S \frac{\partial^2 w(x, t)}{\partial t^2} = 0 \quad (1.3)$$

The beam has a length L , second moment of the cross section I and Young's modulus E . M is the concentrated mass, M_t is the mass of the beam, ρ is the density of the material and S is the cross-sectional area. We assume that the beam is subjected to an unbalanced force applied to the concentrated mass, through a cyclic force $F(x, t)$, with:

$$F(x, t) = p_0 \cos(\Omega t) = m e \Omega^2 \cos(\Omega t) \quad (1.4)$$

Where, p_0 is the excitation amplitude, m is the imbalance mass, e is the imbalance radius and Ω is the rotational speed. Solution to equation 1.3 must comply with the following boundary conditions:

$$w(0, t) = 0; \frac{\partial w(0, t)}{\partial x} = 0; \frac{\partial^2 w(L, t)}{\partial x^2} = 0; -[-EI \frac{\partial^3 w(L, t)}{\partial x^3}] = M \frac{\partial^2 w(L, t)}{\partial t^2} - p_0 \cos(\Omega t) \quad (1.5)$$

A separation of variable method is followed to find the solution of equation 1.3, which gives:

$$w(x, t) = W(x)T(t) = W(x)\cos(\omega t) \quad (1.6)$$

By substituting equation 1.6 into equation 1.3, we obtain:

$$\frac{W(x)^{(IV)}}{W(x)} - \frac{\rho S}{EI} \Omega^2 = k^4 \quad (1.7)$$

By assuming the solution of $W(x)$ in the form:

$$W(x) = A\cos(kx) + B\sin(kx) + C\cosh(kx) + D\sinh(kx) \quad (1.8)$$

where, \sinh and \cosh are hyperbolic functions, and A, B, C and D are to be determined from the boundary conditions of the beam with:

$$k^2 = \omega \sqrt{\frac{\rho S}{EI}} \quad (1.9)$$

The fundamental angular frequency is:

$$\omega_1 = \frac{\alpha_1^2}{L^2} \sqrt{\frac{EI}{\rho S}} \quad (1.10)$$

where, α_1 is the fundamental eigenvalue, it depends on the value of the mass ratio $r = \frac{M}{M_t}$.

Applying the boundary conditions:

$$W(0) = 0; \left. \frac{dW(0)}{dx} \right|_{x=0} = 0; -EI \left. \frac{d^3 W(x)}{dx^3} \right|_{x=L} = p_0 + M\Omega^2 W(L) \quad (1.11)$$

The deflection expression gives the following relation for the beam deflection amplitude:

$$W(x) = -\frac{\frac{p_0 L^3}{EI}}{\beta_1 \alpha_1^3 (\frac{\Omega}{\omega_1})^{3/2} + \beta_2 m \alpha_1^4 (\frac{\Omega}{\omega_1})^2} \left(\sin(kx) - \sinh(kx) - \frac{\sin(kL) + \sinh(kL)}{\cos(kL) + \cosh(kL)} (\cos(kx) - \cosh(kx)) \right) \quad (1.12)$$

with:

$$\beta_1 = \left(-\frac{\sin(kL) + \sinh(kL)}{\cos(kL) + \cosh(kL)} (\sin(kL) - \sinh(kL)) - \cos(kL) - \cosh(kL) \right) \quad (1.13)$$

and:

$$\beta_2 = \left(\sin(kL) - \sinh(kL) - \frac{\sin(kL) + \sinh(kL)}{\cos(kL) + \cosh(kL)} (\cos(kL) - \cosh(kL)) \right) \quad (1.14)$$

1.3.1.2 Analytical model of a rotating flexible beam

The rotating beam considered in this contribution is assumed to be a prismatic cantilever of length L with a hollow rectangular cross section as shown in figure 1.48.

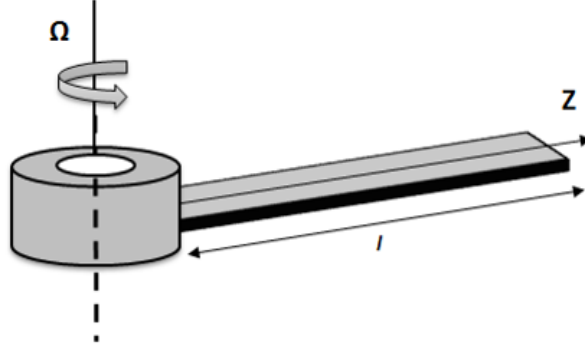


Figure 1.48: Flexible rotating beam

Depending on the Euler-Bernoulli theory, the governing differential equation for the rotating beam under flap wise vibration is given by [Arrigan et al., 2011]:

$$\mu(z) \frac{\partial^2 u}{\partial t^2} + \frac{\partial^2}{\partial z^2} \left(E_b I_b \frac{\partial^2 u}{\partial z^2} \right) - \frac{\partial}{\partial z} \left(T(z) \frac{\partial u}{\partial z} \right) = p(z, t) \quad (1.15)$$

The centrifugal tension force $T(z)$ at a distance z from the origin is given by:

$$T(z) = \Omega^2 \int_z^L y \mu(y) dy \quad (1.16)$$

where, u is the relative movement of a point with respect to its static deflected location. The applied force per unit length p is the unit mass of the beam per length, Ω is the rotational speed of the beam, which is assumed to be constant, E_b is the modulus of elasticity, I_b is the area moment of inertia of the beam cross-section. The impact of gravity on the rotation of the beam is considered to be negligible compared to centrifugal impact.

The clamped and free boundary conditions for the rotating beam are given as follows:

$$u = \frac{\partial u}{\partial z} = 0 \text{ at } z = 0; \quad \frac{\partial^2 u}{\partial z^2} = \frac{\partial^3 u}{\partial z^3} = 0; \text{ at } z = L \quad (1.17)$$

The mode summation method is considered for the solution, that is :

$$u(z, t) = \sum_{i=1}^n \Phi_i(z) q_i(t) \quad (1.18)$$

where, $\Phi_i(z, t)$ are the shape functions and $q_i(t)$ are the generalized coordinates. The shape functions must be linearly independent and must satisfy the geometric

boundary conditions of the hub.

The generalized mass matrix M is defined by:

$$M = \int_0^L \mu(z) \Phi(z) \Phi^T(z) dz \quad (1.19)$$

The potential energy of the system is:

$$U_d(q) = \frac{1}{2} q^T(t) K q(t) - q^T(t) F(t) \quad (1.20)$$

where, K is the generalized stiffness matrix of the blade which is: $K = K_e + K_g$, where K_e is the elastic stiffness matrix, defined by:

$$K_e = \int_0^L EI(z) \frac{d^2 \Phi(z)}{dz^2} \left(\frac{d^2 \Phi(z)}{dz^2} \right)^T dz \quad (1.21)$$

and K_g is the geometric stiffness matrix defined by:

$$K_g = \int_0^L T(z) \frac{d\Phi(z)}{dz} \left(\frac{d\Phi(z)}{dz} \right)^T dz \quad (1.22)$$

The dynamic load vector $F(t)$ is defined by:

$$F(t) = \int_0^L \Phi(z) p(z, t) dz \quad (1.23)$$

The dynamic model is formulated using the Lagrangian formulation expressed by:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}^T} \right) - \frac{\partial L}{\partial q^T} = F(t) = -C\dot{q} \quad (1.24)$$

where,

$$L(\dot{q}, q) = T_d(\dot{q}) - U_d(q) = \frac{1}{2} \dot{q}^T M \dot{q} - \frac{1}{2} q^T K q + q^T F(t) \quad (1.25)$$

By substituting the equation 1.25 into the Lagrangian formulation 1.24, we obtain:

$$M\ddot{q} + C\dot{q} + Kq = F(t) \quad (1.26)$$

where, \ddot{q} , \dot{q} , q are the acceleration, velocity and displacement vectors and C is the damping matrix.

The beam is considered to be vibrating in its fundamental mode and a quadratic mode shape is assumed. This allows the continuous beam to be reduced to a single degree of freedom (SDOF). The bending motion of the beam in the ' yz ' plan is only of interest, where this motion is called flapping motion.

The relative displacement of the beam $u(z, t)$ can be expressed as:

$$u(z, t) = \phi(z) q(t) \quad (1.27)$$

The shape function is taken as:

$$\phi(z) = \frac{3}{2} \left(\frac{z}{L} \right)^2 \left(1 - \frac{1}{3} \frac{z}{L} \right) \quad (1.28)$$

The generalized mass matrix and stiffness matrix are then:

$$M = \frac{3}{140}\mu L; K_e = 3\frac{EI}{L^3}; K_g = \frac{81}{280}\mu L\Omega^2 \quad (1.29)$$

The angular eigen-frequency at $\Omega = 0$ can be expressed as:

$$\omega_{1,0} = \sqrt{\frac{K_e}{M}} = \sqrt{\frac{140}{11}}\sqrt{\frac{EI}{\mu L^4}} = 3.5675\sqrt{\frac{EI}{\mu L^4}} \quad (1.30)$$

and the angular frequency at $\Omega > 0$ becomes:

$$\omega_1 = \sqrt{\left(\frac{K_e}{M} + \frac{K_g}{M}\right)} = \omega_{1,0}\sqrt{\left(1 + \frac{K_g}{K_e}\right)} = \omega_{1,0}\sqrt{\left(1 + \frac{27}{22}\frac{\Omega^2}{\omega_{1,0}^2}\right)} \quad (1.31)$$

We consider that the applied load is a triangular harmonic distributed by unit, the frequency of this sinusoid being equal to the rotation frequency as shown in figure 1.48. This load can be expressed as follows:

$$p(z, t) = p_0(t)\frac{z}{L} = p_0\cos(\Omega t)\frac{z}{L} \quad (1.32)$$

For a zero initial condition and without taking into account the damping effect, the analytical solution in steady state is then :

$$u(z, t) = \frac{11Lp_0}{40M(\omega_1^2 - \Omega^2)}\cos(\Omega t) \times \frac{3}{2}\left(\frac{z}{L}\right)^2\left(1 - \frac{1}{3}\frac{z}{L}\right) \quad (1.33)$$

1.3.1.3 Implementation of the flexible structures in Modelica and application to case-study of a Wind Turbine

The previously developed equations for a flexible beam with a mass and a rotating flexible beam were implemented using the object-oriented and equation-based language Modelica. Every model is encapsulated with a Modelica object and the coupling between the components is ensured with connectors. Modelica connectors are instances of connector classes, which are used to connect the models of components to each other to define a model of a complex system.

Two kinds of variables are used in connectors: *Non-flow* variables which are also called *potential* variables, and *flow* variables which represent some kind of flow.

Modelica generates automatically the coupling equations when components are connected:

- Equality coupling, for non-flow variables
- Sum-to-zero coupling, for flow variables

The value of a flow variable is positive when the flow is into the component.

Frame connector of the Multibody Modelica library was used in this work for the two components: flexible beam with a mass and the rotating flexible beam. Frame

connectors is a definition of a coordinate system that is fixed to a mechanical component. The connector allows the definition of the position and orientation vectors of the component in relation to the absolute frame (World). It allows also the definition of the cut-force and cut-torque vectors resolved in the connector frame. Figure 1.49 shows the implemented Modelica components of the flexible beam with a mass (on the left) and the rotating flexible beam (on the right). The figure shows also the connectors used to allow the coupling of the components to each other or to the other mechanical components of the Modelica library.

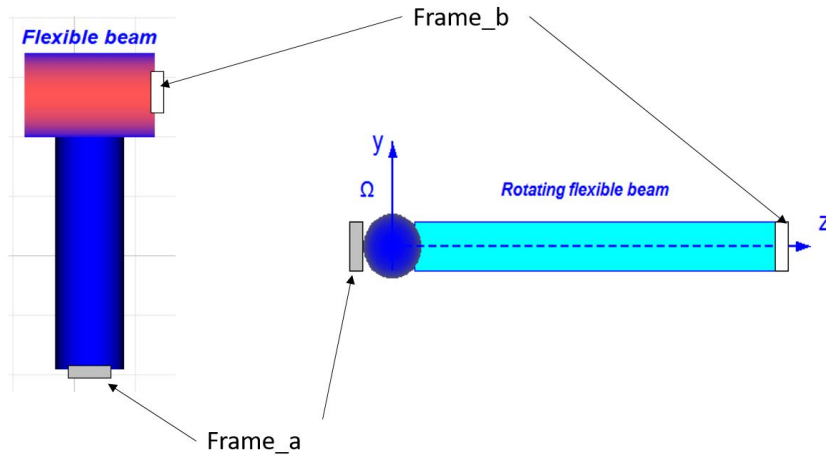


Figure 1.49: Implementation of the Modelica components with their connectors: flexible beam with a mass (left), rotating flexible beam (right)

To validate the modeling approach a case-study of the vibration analysis of a wind turbine was considered. It is known that wind turbines operate in hostile environments. The variable loads with an elastic structure create an ideal ground for induced vibrations. The wind turbine structure is idealized by replacing it with an assembling of a flexible tower with a rigid mass at the top representing the nacelle, and the blades modeled with rotating beams. It will be assumed that the tower is subjected only to a load which represents the effect of rotor imbalance. The Modelica model of the study is illustrated by the figure 1.50.

In this analysis, the rotor is subjected to a thrust force caused by the wind, which in turn is transferred by the nacelle component on the top of the tower. The deformation of the blade and the tower occurs in the same plane. This model thus allows to study the dynamic interaction between the tower and the blade.

For the validation purpose, the vibration analysis results generated by the analytical models developed in this study and implemented with Modelica were compared to the analysis results generated using the components of the commercial FlexibleBodies Modelica library developed by BAUSCH-GALL GmbH and Dassault Systèmes in Dymola software. Figure 1.51 shows a superposition of the nacelle vibration am-

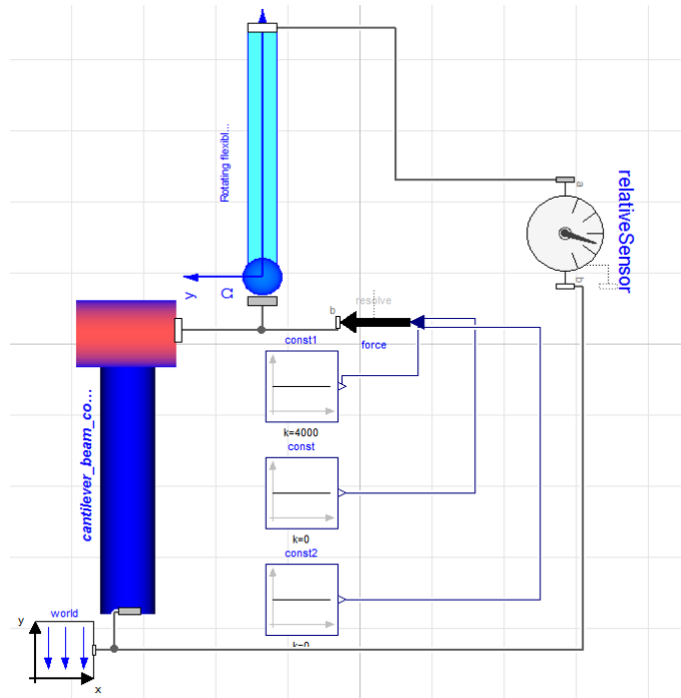


Figure 1.50: Modelica model of wind turbine for the vibration analysis

plitudes generated by the two models.

The comparison of the results shows a strong difference between the two models in the transient phase which is 33%, but it is only 5% once the steady state regime is established. Since the purpose of the model is to be used in the preliminary design phase allowing the quick analysis of the interactions of the different disciplines and physics of mechatronics, it can be stated that the developed models allow to provide this utility to mechatronic designers to make the right decisions before embarking on detailed studies with expensive 3D CAD models and finite element calculations. Finally, the proposed methodology of developing compact models using analytical approach with Modelica implementation can be easily applied to other physics such thermal and electromagnetic compatibility.

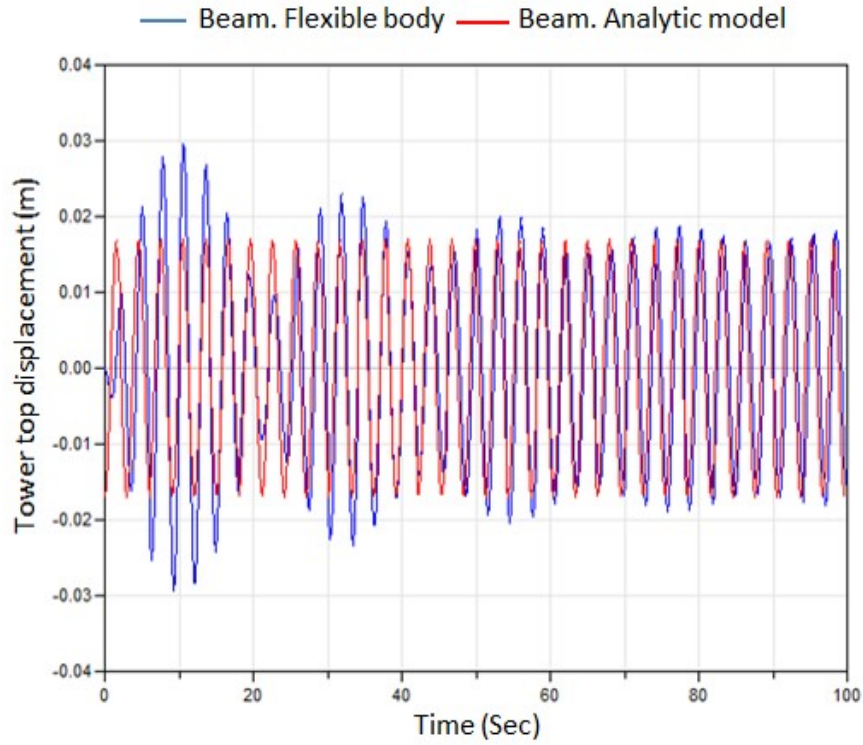


Figure 1.51: Nacelle vibration amplitudes

1.4 Interdisciplinary collaborative design of mechatronic systems

Mechatronic design requires effective collaboration of multidisciplinary teams to share and exchange skills and knowledge. This requires, on the one hand, some kind of database to capitalize knowledge and its instantaneous reuse, but also to develop software means to assist designers during the design process.

In this context, we will present in this section two contributions. The first one concerns the development of a methodology allowing to formalize the knowledge related to both the mechatronic product and its design process. The second contribution concerns the development of a collaborative design approach where designers are assisted by a multi-agent system.

1.4.1 Collaborative Design Process and Product Knowledge (CDPPK) Methodology

This contribution is part of the thesis of Mehdi MCHAREK [Th3] about the knowledge management of the mechatronic design: product and process.

Indeed, the success of any collaboration framework is strongly linked to the need to share knowledge between actors to ensure a common representation of the problem to be solved. However, the shared knowledge is composed of fragments that are

created by various actors according to their expertise domain. An important aspect to be considered is then the adaptation to these different viewpoints between three types of actors : System engineer or system architect, and specialists in simulation and optimization.

One may wonder how we can adapt the already available knowledge configuration management (**KCM**) approaches in a way that covers the entire design cycle. Our goal is to manage and communicate knowledge between the decomposition phase and the integration phase of the development cycle of a mechatronic product. Our framework aims at overcoming collaboration and reuse issues. For collaboration, a new way of viewing the product is necessary to find solutions that lie across disciplinary boundaries. For the reuse, the previously proposed **KCM** model, presented in section 1.1.2.6, focuses on declarative knowledge (know-what). It is important that this knowledge helps to understand the product, but it must be complemented by process intentions (know-how) to facilitate the reuse of a design project.

The new **KCM** model that we proposed, illustrated in figure 1.52, is inspired by **KCM** structure and the new contribution lies firstly on reorganizing parameters in order to facilitate collaboration. In fact, all parameters involved must be reflected in a well formalized conceptual infrastructure to ensure an effective collaboration between the system architecture (SA) and specialists in modeling and simulation (MS). On the one hand, we propose the use of **ICEs** as a point of reference for SA actors. On the other hand, MS actors have only to instantiate the parameters that they need from specific **ICEs**. Subsequently, the MS engineers become detached from the conceptualization task of the **ICEs**.

Additionally, for each parameter an in/out flow attribute is added to facilitate the connection between the **UCs** of MS engineers, and create a collaborative network that will be used later for MS-**MDAO** collaboration. The main concepts of the new model and the methodology to connect SA-MS and MS-**MDAO** will be detailed in the next sections.

1.4.1.1 CDPPK Main concepts

Information Core Entity (ICE)

The concept of **ICE** is kept in this methodology to capitalize knowledge. The difference with the previous methodology presented in the background section 1.1.2.6 is that the **ICE** is considered here as an organizational structure only for the decomposition phase. System engineers use **ICE** to decompose the system into functional subsystems and save the associated parameters and constraints. Therefore, **ICE** is used here to transform product data into product information. The parameters can be: reals, intervals, vectors and matrices to give more flexibility to the stakeholders. For the constraints, we consider only interval constraints in

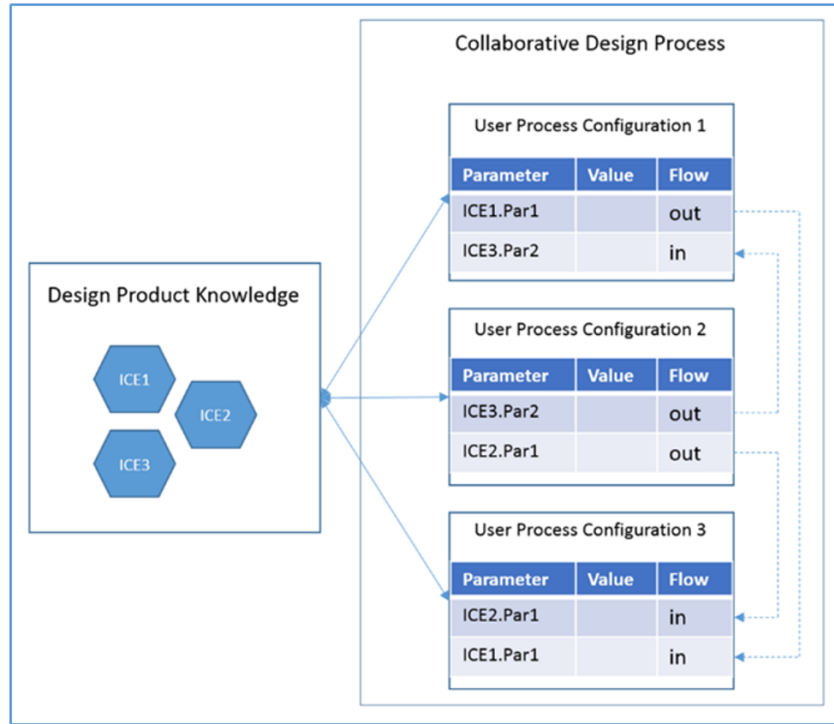


Figure 1.52: Collaborative Design Process and Product Knowledge principle

this work.

Design Product Knowledge (DPK)

Design Product Knowledge (DPK) contains all the ICEs used in the project. Stakeholders can add here the parameters and the constraints. DPK can be considered in this stage as a product information base. When the collaboration starts, stakeholders refer to this base which represents the system level. DPK also plays an important role in conflict management, as we will explain later. After the collaboration, ICEs will contain the final product results. Thus, DPK generates product knowledge from information stored in ICEs.

Usage Process Configuration (UPC)

This new class differs from the UC defined in KCM. Instead of affecting a block of parameters in the form of ICE instances, we propose to break ICEs and give the flexibility to disciplinary engineers in the choice of parameters that they need. Our contribution is to define input/output flows for the instantiated parameters. This considerably facilitates the collaboration and data exchange between stakeholders and helps to construct a collaboration network.

Collaborative Design Process (CDP)

It groups the Usage Process Configuration (UPC)s present in the collaboration.

In Collaborative Design Process (CDP), we can generate a global collaboration process. It characterizes the originality of this approach by achieving the traceability of the collaboration evolution for reuse perspective. This collaboration graph presents the design process knowledge.

Project Domain (PD)

The concept of Project Domain (PD) makes it possible to define a macroscopic context for the definition of the basis of ICEs and the use of configurations. Indeed, we can qualify a PD as a global usage environment. For example, a system can be defined in PD and can be decomposed into major subsystems. In this way, we capitalize the respective knowledge to each subsystem and reduce complexity in collaboration and traceability. Each Knowledge Domain has its own base of ICEs and configurations to facilitate the reuse.

CDPPK metamodel

This new model is adapted to the decomposition phase. Each functional subsystem is represented by an ICE. Then, in the integration phase, each specific domain tool is represented by a UPC. We notice that the knowledge cycle (data-information-knowledge) for the design process and product is respected. The UML metamodel of Collaborative Design Process and Product Knowledge (CDPPK) framework is shown in figure 1.53.

The different classes and connections are represented between the entities defined previously. This model is based on the concept of Configuration Management (CM) defined in ISO 10007:2015. CM carries the access to the adequate information relative to product design, realization, verification and use. This information is organized in a “global configuration” which is updated and versioned to trace the evolution of the product. In CDPPK, this organizational concept is the Knowledge Configuration (KC). The standard defines also “configuration items” that are components of the global configuration to ensure the final use of information and allow users to converge to a single product. In our model, these items are the UPCs.

CDPPK process

Actors in the design cycle are submerged in a heterogeneous tools environment and information flow. They collaborate with traditional and informal ways (email, meeting...). These ways are error-prone and difficult to trace for reuse perspectives. Whereas, if we have effective collaboration we can acquire not only short-term benefits, during the product development, but also long-term ones, during the reuse process. Knowledge Management (KM) was proposed to improve collaboration performance in mechatronic design. Nevertheless, it is not limited to its technological aspect for it, also, includes a sociotechnical facet which considers knowledge as a resource participating in companies’ performance [Arduin et al., 2013]. A very common problem of a failed collaborative framework is that it does not meet the users’ real tasks. Therefore, the CDPPK’s methodology is proposed with regards

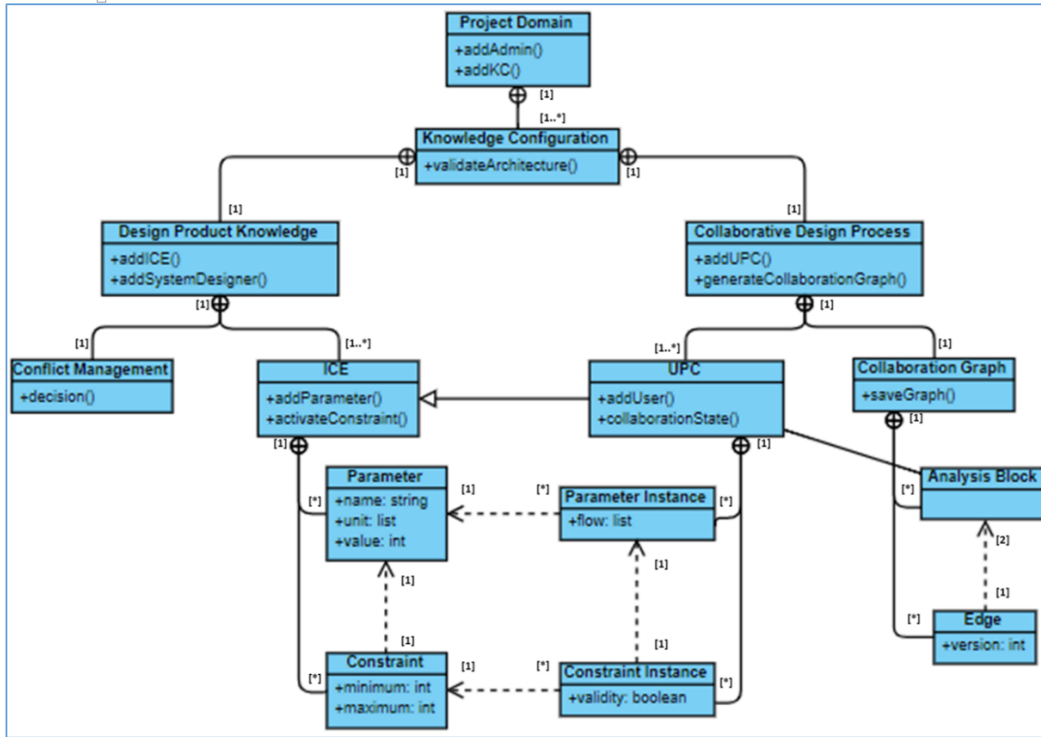


Figure 1.53: UML metamodel of CDPPK

to the mechatronic design cycle, as illustrated in figure 1.54.

Each step is adapted to each stakeholder involved in the design cycle and their needs.

The process is summarized as follows:

Step1: Project Requirements:

The project manager analyses the customer requirements. Accordingly, they create a knowledge configuration **KC** and identify the actors needed for each step. Their main objective is to lead the team towards accomplishing the project.

Step2: Conceptual Design:

Based on customer requirements, system engineers define the functions of the system. They can explore several architectures according to the defined functions, in this work, we will focus on the development of one architecture. Our hypothesis is to decompose the system into functional subsystems. Each one of the latter is represented by an **ICE**. The available parameters and constraints (from requirements and previous works) are added in this phase to the **ICEs**.

Step3: Detailed Design:

Here, the previously created **ICEs** are enriched with all the crucial parameters necessary for the integration phase. To do so, we recommend a meeting between the different stakeholders. A knowledge expert can also capitalize the crucial knowledge necessary for the collaboration. Semi-automatic extraction of parameters methods exists also. This step is a direct intermediate between the decomposition and the

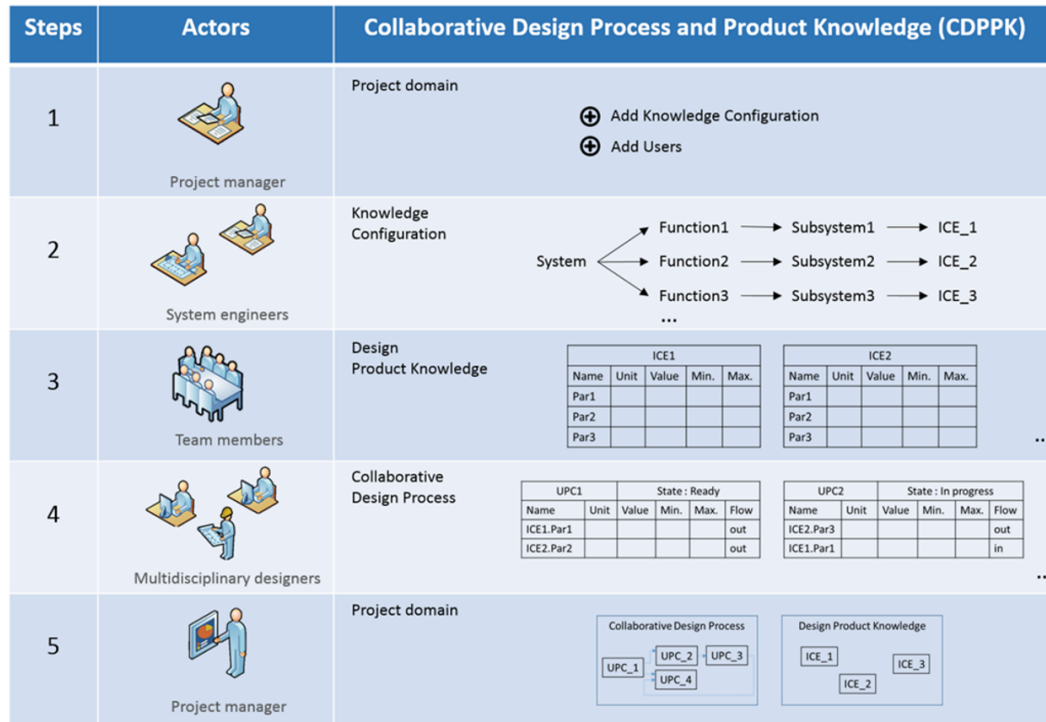


Figure 1.54: Collaboration methodology between System architect and MS actors with CDPPK

integration phases.

Step4: Verification & Integration:

In this step, disciplinary engineers create their own User Configuration Process (UPC). Each engineer instantiates the parameters that they need in their own UPC. Then, they instantiate the parameters to link with their disciplinary domain models. Then, the in/out flow is selected for each parameter. When all the engineers have filled their UPC, the collaboration starts and the exchange of the parameters follows the network generated by different parameters flows. Each designer gets to know which UPCs are currently related to their work. Each UPC has a collaboration state to manage the exchange (Start/Ready/In Progress/Publish).

Step5: System Validation:

Thanks to the product knowledge in DPK and the process knowledge in CDP, the project manager has a holistic view of the design cycle. They need to check the collaboration state and the recorded conflicts between system engineers and disciplinary designers in order to assess the situation and make decisions.

This methodology enables each stakeholder to access and use the right knowledge. In managerial terms, our methodology moves from the traditional predefined collaboration process to a more dynamic and flexible one. It provides a common basis of exchange that does not disrupt the tasks of the stakeholders. The main dis-

advantage of such a methodology is the need to bring the different stakeholders together to formalize the knowledge, which can be very time consuming.

1.4.1.2 Collaboration using **CDPPK** and management of conflicts

According to our methodology, conflicts are generated when parameters' values do not respect the constraints. The constraints associated with the parameters can be system constraints (defined in an **ICE**) or interdisciplinary constraints (defined in **UPC**). Therefore, system constraints help us to manage requirements' changes and interdisciplinary constraints help us to manage conflicts between disciplinary engineers (figure 1.55). We defined collaboration states for **UPC** and **CDP** in order to organize the conflict resolution process.

The purpose of conflict management is to detect inconsistencies and trace this

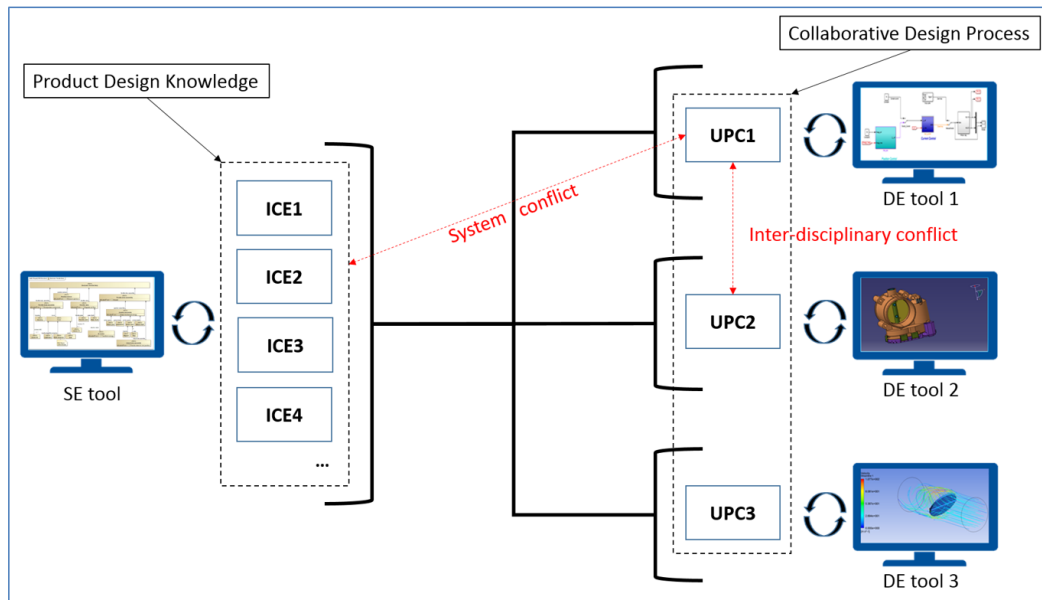


Figure 1.55: System and interdisciplinary conflicts management in **CDPPK**

information back to users to allow them to iterate, make decisions and finally converge towards a common solution. Once the conflicts are detected, it is a matter of putting the information back to the different users. In the case of interdisciplinary conflicts, the conflicts are listed in **CDP** and managed directly by disciplinary engineers. In the case of system conflicts, it is the **DPK** that is responsible for displaying this information for system engineers. In this way, the traceability and coherence of manipulated knowledge is guaranteed in the entire cycle.

1.4.1.3 Validation of the **CDPPK** methodology

For the validation purpose, we developed a demonstrator using Python language to implement the **CDPPK** model. The figure 1.56 shows the implementation of **ETB**

ICEs in the CDPPK demonstrator. We developed the different simulation models

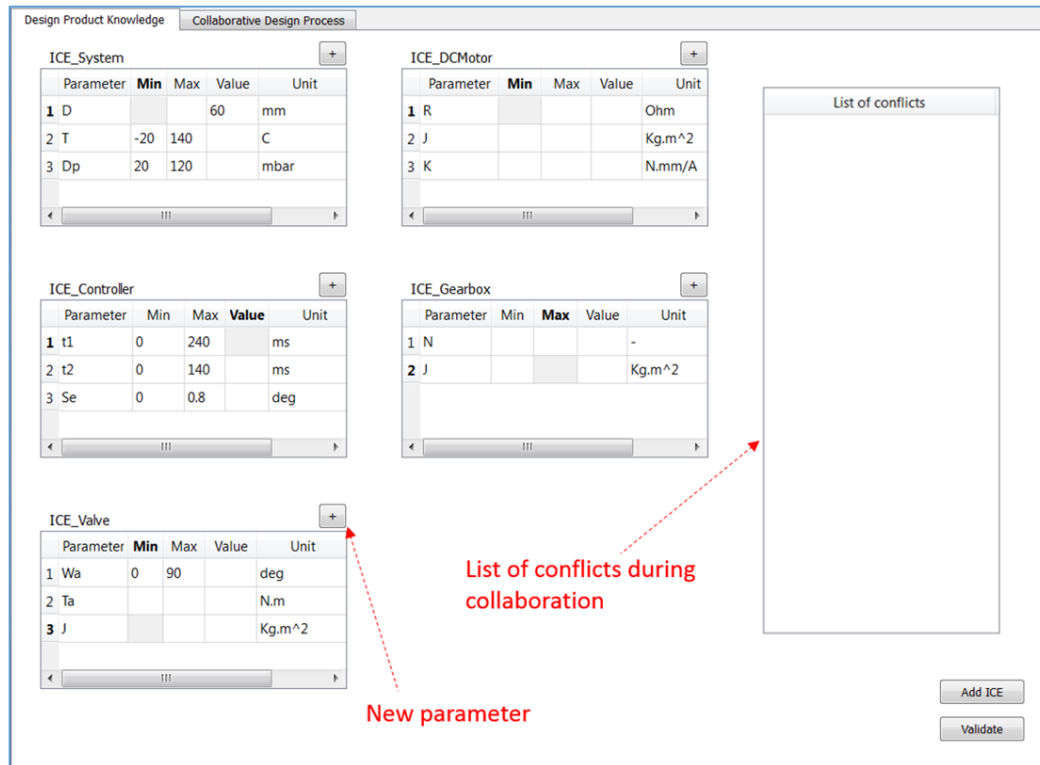


Figure 1.56: ICEs definition in DPK menu of the CDPPK model

used for the Electronic Throttle Body (ETB) design cycle as well as a test bench was constructed for the validation purpose. Based on industrial requirements we used our demonstrator to manage the design cycle. Figure 1.57 shows the UPCs used for the collaborative design process of the ETB case-study defined. The figure 1.58 shows the implementation of the collaborative design process (CDP) in the demonstrator. To sum up, with the CDPPK model, knowledge generated by the collaborative activity can be stored into product knowledge and design process knowledge in a way that it can be easily reused. On one side, the product knowledge serves as a source of information for system engineers and disciplinary engineers. The system engineers use this knowledge to decompose the product parameters and requirements. This is suitable for their system view. The disciplinary engineers select the parameters that they need to collaborate to search the optimal values. On the other side, the dynamic process knowledge breaks with the traditional predefined design process which is not suitable for concurrent engineering.

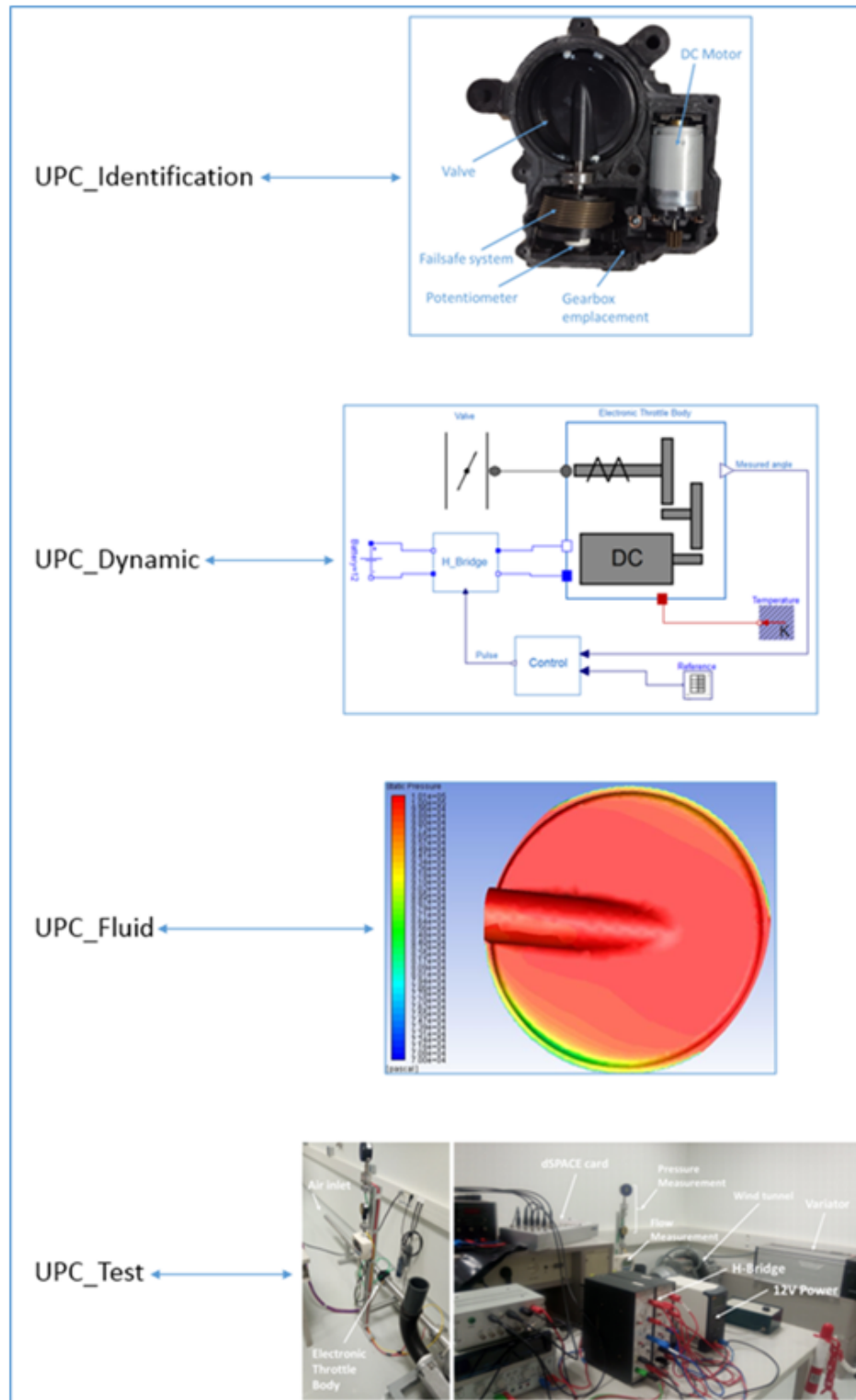


Figure 1.57: The different UPCs involved in the collaborative scenario of the ETB design

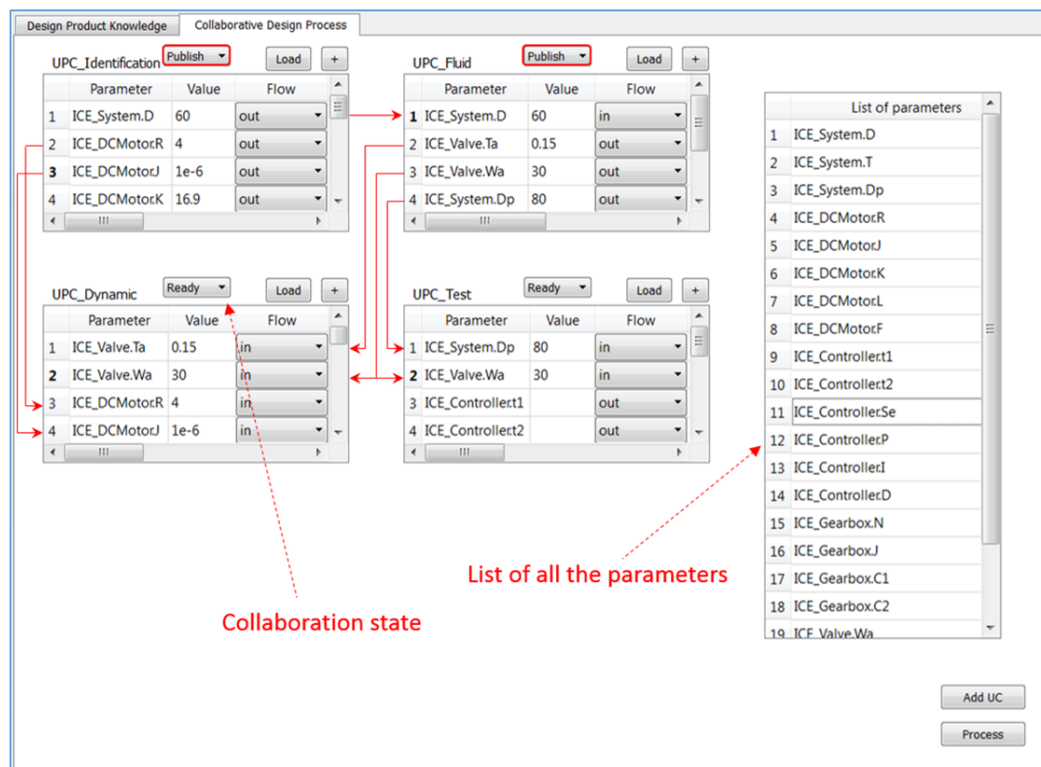


Figure 1.58: implementation of the collaborative design process (CDP) in the CDPK demonstrator

1.4.2 Collaborative design assisted by multi-agent systems

This contribution is part of the thesis of Amir GUIZANI [Th1] about a collaborative design approach, combining designers and virtual agents, to collectively find optimal solutions that satisfies the constraints and performance requirements of a mechatronic design. The proposed approach is based on multi-agent system. The objective is to adapt this technique with the distributed collaborative design to benefit from its properties. We start by recalling the definition of multi-agent systems and their properties to detail the contribution.

Several definitions attributed to the notion of agent exist in the literature. However, the definition of the term agent proposed by Ferber and Perrot [Jacques, 1995] is widely used in the scientific community. Ferber and Perrot define an agent as being a physical (hard) or software (soft) entity located in an environment (real or virtual) which:

- is able to act in its environment,
- can communicate with other agents of the same platform or different platforms,
- is able to perceive its environment in a limited way,
- has its own resources,
- has skills
- offers services,
- have a partial view of its surroundings,
- can possibly recur.

The definition which was given later by [Jennings et al., 1998] is: "an agent is a computer system, located in an environment and which acts in an autonomous and flexible way to achieve the objectives for which it was designed".

In order to better understand the notion of agent, we can identify several properties that can be attributed to agents [Wooldridge and Jennings, 1999]:

- **Autonomy:** the ability of an agent to act alone to achieve these objectives, and is not subject to any control over his internal state, nor over the actions it performs.
- **Social capacity:** the ability of an agent to interact with other agents and possibly with humans through communication languages.
- **Reactivity:** the ability of an agent to perceive its environment and to respond in a timely manner to the various changes that occur in it.
- **Adaptability:** the ability of an agent to learn and improve with experience.

- Pro-activity: the agents must not react only to stimuli coming from their environment, they must also be able to show behaviors directed by internal objectives and this by taking initiatives.
- Continuous over time: it is a process that is continuously in execution.

Many software platforms exist for the development of multi-agent systems. The choice of a platform depends on how easy it is to implement the different types (models) of agent and how easy it is to use. Indeed, the internal characteristics of the agents are more or less easy to implement depending on the development platform used. Among the best known platforms, we find: CORMAS, DoMIS, JACK, JADE, Jadex, Janus, Madkit, etc.

JADE for Java Agent DEvelopment is an open source Multi-Agent System (MAS) development framework based on the Java language. It offers in particular an advanced support of the FIPA-ACL standard, as well as tools for syntactic validation of messages between agents based on ontologies.

In our case, we chose the Java Agent Development (JADE) platform [Bellifemine et al., 1999] as a development environment for the collaborative design platform. JADE includes all components to manage the collaborative platform according to the Foundation for Intelligent Physical Agents - Agent communication Language (FIPA-ACL) standard. These components are: Agent Management References Model (AMRF), Message Transport System (MTS), Agent Management System (AMS) and Directory Facilitator (DF).

The JADE application is deployed on one or more machines. It hosts a set of uniquely identified agents that can communicate bidirectionally with other agents. Each agent runs in a secondary container that provides its execution environment; it can migrate within the platform. Each platform must have a main container (mainContainer).

Figure 1.59 represents a JADE application developed on 3 machines to assist the collaboration of two designers. The coordinating agent CA is deployed in the main container (mainContainer), and the design agents DA1 and DA2 are deployed in container 1 and container 2.

The framework for the agent-based collaborative design is shown in figure 1.60. The agent platform (AP) is the software environment in which multiple agents can be created, deployed, registered, located, and communicate. Each agent must be registered with the Agent Management References Model (AMRF) descriptions, which provide a standard infrastructure with well-defined functions, to provide specific services. The agent management system (AMS) is a registry directory that assigns a unique identifier to each agent (white page service: automatically references agents by name when they enter the system). The directory facilitator (DF) is a registry directory that provides the yellow pages service (referencing agents

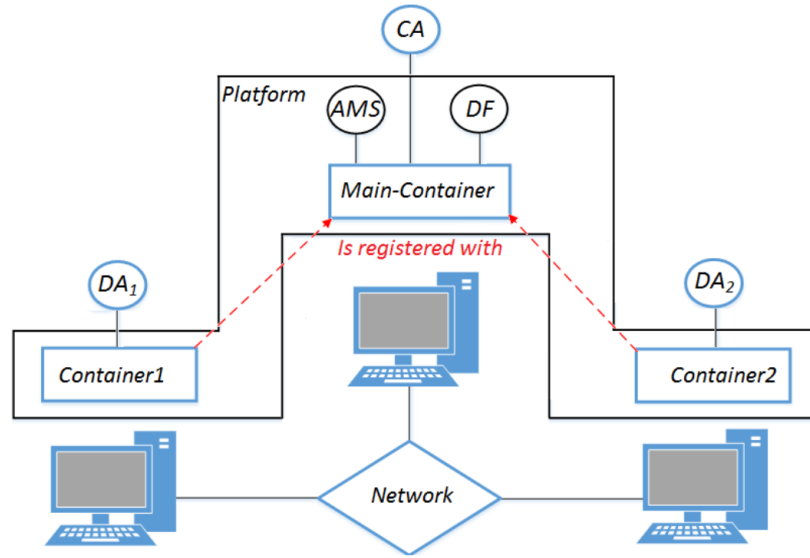


Figure 1.59: Case of collaboration of two design agents through a JADE platform [Guizani, 2016]

by their service(s)). Agent's services can be queried by the information stored in DF. The message transport system (MTS) coordinates the communication between agents of the same or different agent platforms.

To the default agents defined in JADE, we added to new types of agents:

- The design agent (DA): which is the piece of software associated to disciplinary team or member participating in mechatronic design.
- The coordinating agent: which is a JADE agent that coordinates the collaboration during a design problem resolution. The coordinating agent could be associated to a real physical designer or a project manager.

The definition of the number of design agents depends on how the multidisciplinary design problem will be decomposed into more solvable sub-problems (partitions). This decomposition can be done according to disciplines, use-cases, design tools, design knowledge, etc.

The coordinating Agent (CA) resides in a primary main-container, and each design agent (DA) is deployed in a secondary container. These containers can be placed separately in different machines and distinguished by different IP addresses.

Coordination protocols specify the sequence of messages between agents. They define the messages to be sent by specifying: the performative, the content, the recipient(s) and establish the responses to these messages. For our approach, the message sent from a coordinating agent (CA) to a design agent (DA) is defined as the formulation of an optimization sub-problem of the design, in which the objectives, the constraints and the design variables are specified. The message sent from a design agent (DA) to the coordinating agent (CA) is a set of possible

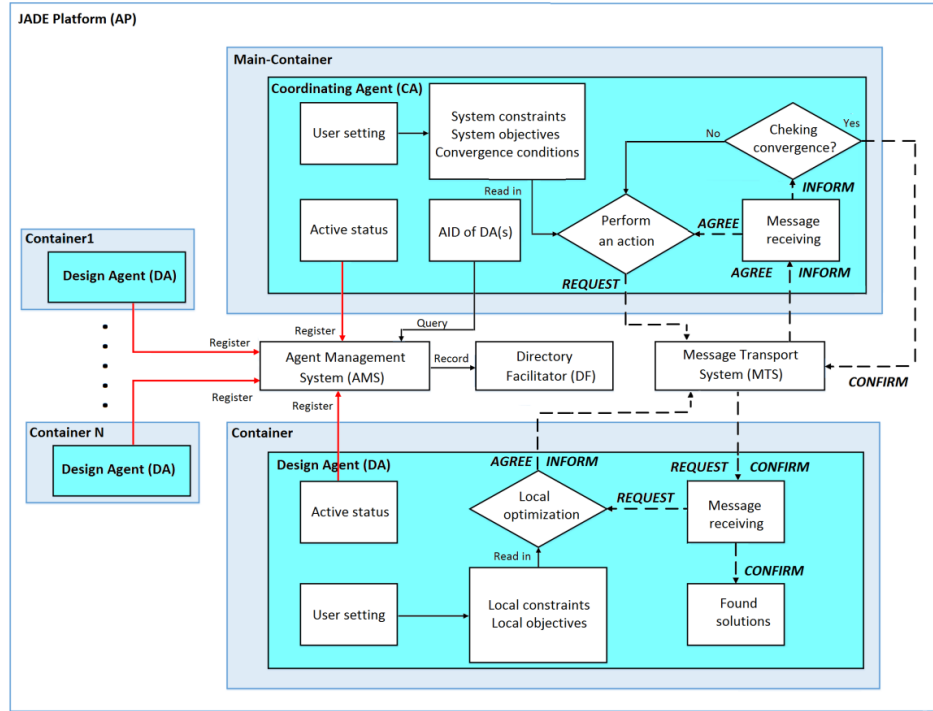


Figure 1.60: The implementation of a JADE platform for collaborative design of mechatronic systems [Guizani, 2016]

solutions. This set could contain many solutions (e.g. Pareto front), one solution, or no solution.

The proposed coordination process is realized in the agent-based system as follows:

1. Initialization: All the design agents needed for coordination are registered in the agent management system (AMS). The information of each design agent is stored in the directory facilitator.
2. Specification: The global objectives, global constraints and convergence conditions of the system are stored in coordinating agent (CA).
3. Information: The coordinating agent (CA) specifies the design agents (DAs) involved in the coordination by querying the agent management system (AMS) and invites all the design agents to join the coordination process.
4. Coordination: the coordination process begins for the determination of the optimal solutions that meet all the requirements requested from the different design agents.

Figure 1.61 represents a communication workflow during a design collaboration between two design agents with a coordinating agent.

The behavior of the agents is implemented using Java language within JADE framework. For instance, the coordination behavior between the agents is carried

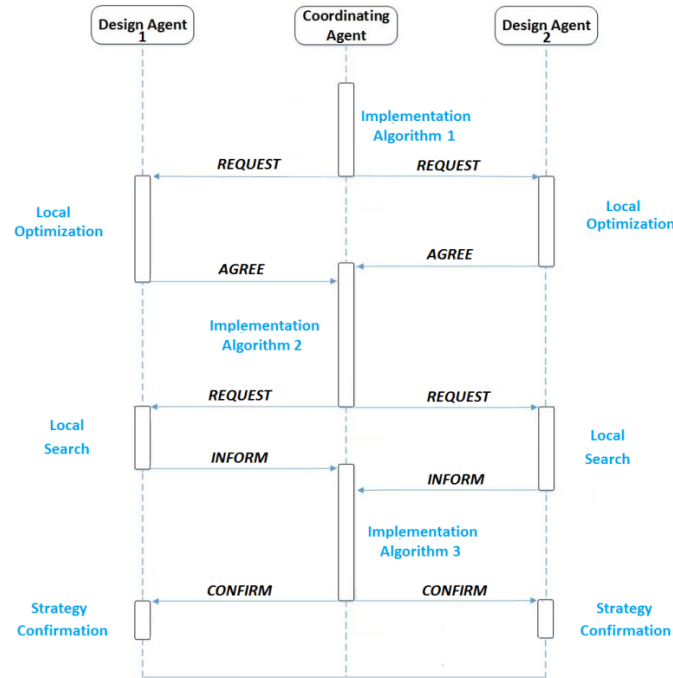


Figure 1.61: Communication workflow between Coordinating Agent and Design Agent [Guizani, 2016]

out according to the algorithm defined in figure 1.62. During the collaboration process, the coordinating agent (CA) informs the design agents (DAs) through the message transport system (MTS) that the communication workflow is starting. The coordinating agent (CA) sends a request by sending *ACLMessage.Request* to each design agent to perform a local optimization and the content of the message is the formulation of the optimization sub-problem of the corresponding agent. Each design agent receives a request to perform a local optimization and determine the optimal solutions with respect to the local objectives in order to satisfy the requested requirements. The optimization itself is made by physical designers. Once the optimization is performed and the solutions are generated, each design agent sends a *ACLMessage.AGREE* message to the coordinating agent. The content of the message is the set of solutions found.

The coordinating agent (CA) waits for the reception of all the messages from the different design agents implied. Then, the coordinating agent (CA) reduced the search design space of the optimal solutions by eliminating the solutions that do not satisfy all the design agent's constraints. A MiniMax algorithm was developed for this purpose (See [Guizani, 2016] for details).

The common solutions found are communicated back to design agents (DAs) to confirm the solutions found with *ACLMessage.CONFIRM* or to follow the collaboration process if required.

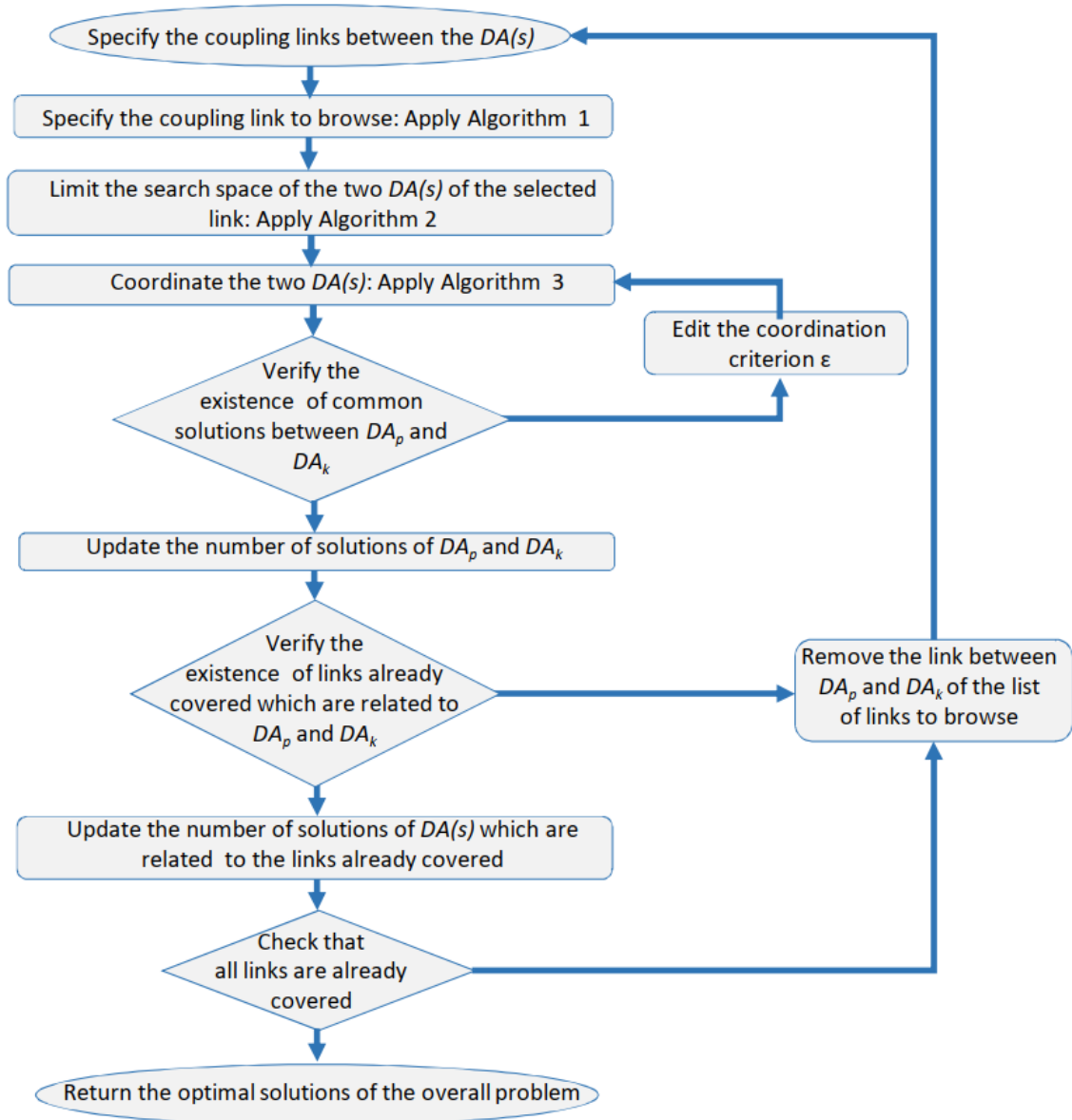


Figure 1.62: MAS coordination algorithm

To validate the approach a case-study of a preliminary design of a propulsion system of an electric vehicle has been considered. The figure 1.63 shows the implementation of the MAS solution with JADE. The design requirements are related to the

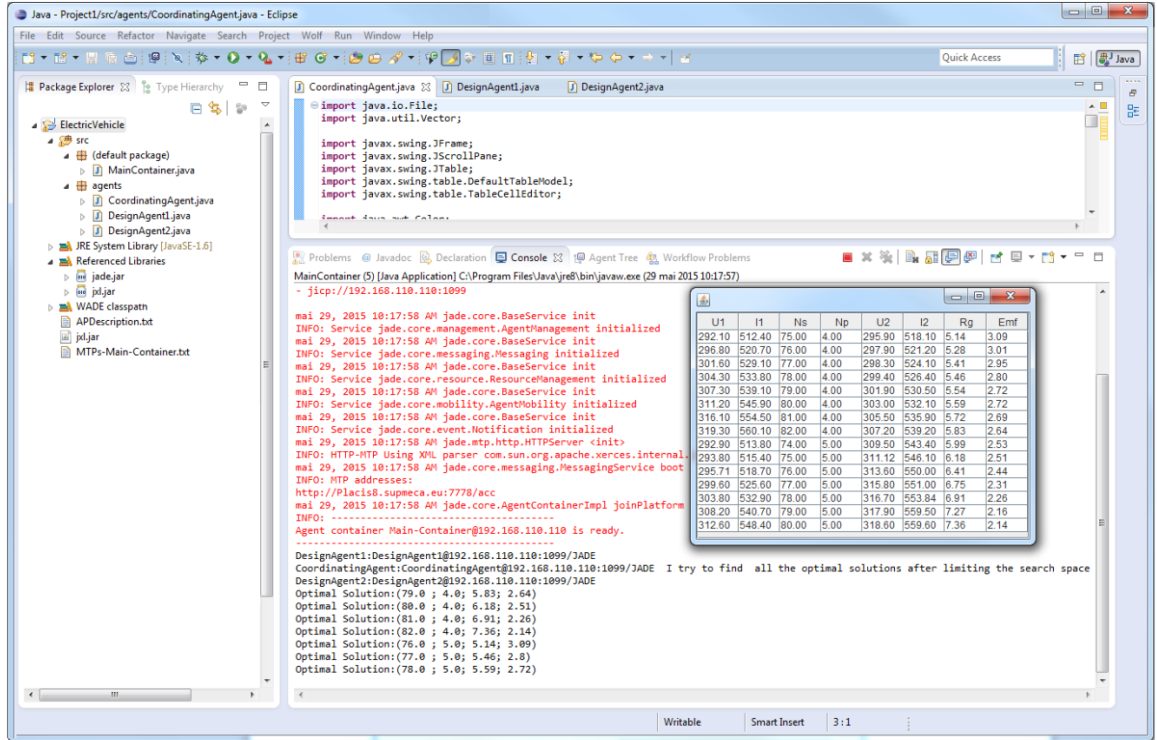


Figure 1.63: MAS implementation

maximum speed of the vehicle, the acceleration performance and the gradeability performance. The design variables are related to the components: battery (number of cells), electric motor (electromotive force) and the gearbox (global gear ratio). The design problem was partitioned into two design sub-problems, one associated to the determination of the battery parameters and the second for determining the other parameters of the propulsion system. Thus, two design agents DA_1 and DA_2 were implemented in JADE with a coordinating agent CA . After solving the two optimization sub-problems, each design agent returns a table of values containing the possible solutions. The coordinating agent then searches for common solutions of the design problem.

To conclude this contribution, a collaborative design methodology based on multi-agent systems has been proposed and a proof of concept has validated the proposal. For this, two types of agents, which are design agent and coordinating agent, were proposed. The difficulty lies in the programming of the agents' behaviors. We proposed to associate a design agent to each design partition. The exchange of messages between the agents during the cooperative solution search phase is done on the basis of a query associated to a formulation of an optimization

sub-problem and the response is a set of design solutions.

As perspectives, we propose to develop a collaborative design platform integrating knowledge management, by extension of the ICE concept, with the multi-agent system, where agents can learn during the collaborative design process, with deep learning for example, which allows virtual agents to assist designers, with the knowledge acquired, during the design process.

1.5 Conclusions and synthesis

In this chapter, I have presented 7 contributions in the field of integrated and collaborative design of mechatronic systems.

The first contribution concerned the co-joint modeling of the mechatronic product and the production process. In this contribution a systems engineering approach supporting the SBCE method has been proposed. The different attributes of both the mechatronic product and its production system are specified and modeled with SysML. The SBCE method allows to cascade decisions according to the availability of information. This allows to explore a large panel of design solutions and system architectures in the early design phases, integrating the production requirements, before deciding on the solution to be detailed.

The second contribution, which deals with the modeling and evaluation of the reconfigurability of manufacturing systems, aims at evolving the MBSE model of a manufacturing system developed with SysML into a simulation model allowing the verification of its **reconfigurability**. Indeed, the reconfigurability of the manufacturing system allows the designer to verify several architectures of a product in early design phases. For this purpose, a new SysML profile has been developed, using the DSL possibilities of UML/SysML, to model the structure of the manufacturing system according to the Adaptive holonic architecture (ADACOR). This holonic architecture is therefore used by the designer to develop a metamodel of the multi-agent system that will be used for simulation. Indeed, holons and agents share various similar attributes allowing to model the reconfigurability of a manufacturing system. The metamodel of the multi-agent system developed in the SysML environment allows, on the one hand, to ensure traceability with the attributes of the manufacturing system and the product to be developed, and on the other hand, it will be used as a support for the implementation of the multi-agent simulation model in an agent-based simulation environment.

In the third contribution, about a MBSE approach for integrating **manufacturability** and **reliability** in the design process of mechatronic systems, a new SysML profile allowing the specification of reliability and manufacturability was developed. Also developed using the DSL possibilities of UML/SysML, this profile diagram allows experts from different domains (manufacturing, reliability, etc.) to

bring their contributions at different design levels according to the SBCE principles. The experts contributions are used to specify the KVAs used for the evaluation of DFX metrics. The evaluation is performed using computational models linking the DFX metrics with KVAs. Thus, a **reliability** assessment model was developed based on the FMEA method.

The fourth contribution, about the integration of **assemblability** in the MBSE approach, A methodology combining a MBSE model based on SysML and 3D CAD model has been proposed. The method consists of 3 phases: a product specification phase with definition of assembly requirements; a 3D design and assembly analysis phase; and a validation of the design solution from the two points of view: **functionality** and **assemblability**. Assemblability evaluation is performed using KVAs: the maximum number of parts, total assembly time (TAT) and assembly efficiency (AE).

The fifth contribution about the development of compact models for vibration analysis of mechatronic systems is linked to the second research theme of multiphysics simulation methods for mechatronic design. In this contribution, analytical solutions of the partial differential equations (PDE) modeling the vibration phenomenon have been deduced and implemented in a Modelica library. The developed models concern a flexible beam fixed on one side and subjected to a mass on the other side, and a flexible beam in rotation. The implementation of the models in Modelica allows the designer to combine them with other Modelica libraries to perform multiphysics simulation including vibration. The approach has been validated using a case study of a wind turbine.

The sixth contribution is about the development of the CDPPK methodology. A new collaboration model integrating model knowledge and knowledge about the design process was developed. The model is based on the use of the concept of ICE but extends it to include information about the design workflow. Indeed, we propose to add inputs/outputs in the ICEs to both compose a knowledge on the product from the data but also a knowledge on the collaboration workflow. The methodology has been validated with a case study of an ETB, for which a test bed has been realized for the experimental validation aspect and a proof of concept has been developed using Python. In this case study ICEs for multiphysics, control and MDAO were used to illustrate the effectiveness of the multidisciplinary collaboration. This work has been realized within the framework of the MIME project and the contribution aims at extending the possibilities of the KARREN software of Digital Product Simulation (DPS) company.

The seventh contribution aims at integrating the use of virtual agent in the collaborative design process of mechatronic systems. For this, a multi-agent system was developed using JADE platform, by adding two types of agents to the default used agents in JADE. The first agent type is a Design Agent (DA) which role

is to receive requests in the form of an optimization sub-problem formulation, and to send a response in the form of a set of solutions defined by an array. The optimisation it self is perfomed by a physical designer by using MDAO for example. The second type of agents is the Collaborative Agent (CA), which role is to send requests to DAs and receive the design solutions. The behavior of an agent is programmed with JAVA. Thus, an algorithm allowing to search for common solutions among the solutions sent by the DAs has been implemented. The algorithm also allows reducing the number of solutions by deleting those that do not satisfy all the design constraints.

Table 1.2 summarizes the contributions presented above and the research topics treated by the PhD students I have co-supervised.

PhD student contributions	MBSE	MSD	MPS	DFX	MDAO	ICDKM
Contrib.1 (R.AMMAR)	X	X		X		
Contrib.2 (A. ABID)	X	X		X		
Contrib.3 (M.F BORCHANI)	X	X		X		
Contrib.4 (R. BRAHMI)	X	X		X		
Contrib.5 (G. HAMZA)		X	X			
Contrib.6 (M. MCHAREK)	X	X	X		X	X
Contrib.7 (A. GUIZANI)		X	X		X	X

Table 1.2: Research topics treated and the PhDs contributions

Our past and current work leads us to consider a certain number of short, medium and long term perspectives in my research activities. All these perspectives are presented in the following chapter.

References

- [Abid, 2019] Abid, A. (2019). *Approche basée sur l'ingénierie système pour la conception intégrée des systèmes manufacturiers reconfigurables*. PhD thesis, Université Paris-Saclay; École nationale d'ingénieurs de Sfax (Tunisie). (Not cited.)
- [Alexandrov, 2005] Alexandrov, N. (2005). Multidisciplinary design optimization. *Optimization and Engineering*, 6(1):5–7. (Cited on page 28.)
- [Arduin et al., 2013] Arduin, P.-E., Grundstein, M., and Rosenthal-Sabroux, C. (2013). From knowledge sharing to collaborative decision making. *International Journal of Information and Decision Sciences*, 5(3):295–311. (Cited on page 82.)
- [Arrigan et al., 2011] Arrigan, J., Pakrashi, V., Basu, B., and Nagarajaiah, S. (2011). Control of flapwise vibrations in wind turbine blades using semi-active tuned mass dampers. *Structural Control and Health Monitoring*, 18(8):840–851. (Cited on page 74.)
- [Bambill et al., 2003] Bambill, D. V., Escanes, S. J., and Rossit, C. A. (2003). Forced vibrations of a clamped-free beam with a mass at the free end with an external periodic disturbance acting on the mass with applications in ships' structures. *Ocean engineering*, 30(8):1065–1077. (Cited on page 72.)
- [Bandler et al., 1994] Bandler, J. W., Biernacki, R. M., Chen, S. H., Grobelny, P. A., and Hemmers, R. H. (1994). Space mapping technique for electromagnetic optimization. *IEEE Transactions on microwave theory and techniques*, 42(12):2536–2544. (Cited on page 32.)
- [Barton and Meckesheimer, 2006] Barton, R. R. and Meckesheimer, M. (2006). Metamodel-based simulation optimization. *Handbooks in operations research and management science*, 13:535–574. (Cited on page 32.)
- [Bedeoui et al., 2019] Bedeoui, A., Hadj, R. B., Hammadi, M., Trigui, M., and Aifaoui, N. (2019). Assembly sequence plan generation of heavy machines based on the stability criterion. *The International Journal of Advanced Manufacturing Technology*, 102(9):2745–2755. (Cited on page 27.)
- [Belhadj et al., 2021] Belhadj, I., Hammadi, M., and Aifaoui, N. (2021). Parallel disassembly approach with recycling rate calculation of industrial products. *The International Journal of Advanced Manufacturing Technology*, 113(9):2969–2984. (Cited on page 27.)
- [Belhadj et al., 2020] Belhadj, I. h., Hammadi, M., Trigui, M., and Aifaoui, N. (2020). Generation of disassembly plans and quality assessment based on cad data. *International Journal of Computer Integrated Manufacturing*, 33(12):1300–1320. (Cited on pages 27 and 28.)

- [Bellifemine et al., 1999] Bellifemine, F., Poggi, A., and Rimassa, G. (1999). Jade—a fipa-compliant agent framework. In *Proceedings of PAAM*, volume 99, page 33. London. (Cited on page 90.)
- [Boothroyd and Alting, 1992] Boothroyd, G. and Alting, L. (1992). Design for assembly and disassembly. *CIRP annals*, 41(2):625–636. (Cited on page 69.)
- [Boothroyd et al., 2010] Boothroyd, G., Dewhurst, P., and Knight, W. A. (2010). *Product design for manufacture and assembly*. CRC press. (Cited on page 24.)
- [Borchani, 2019] Borchani, M. (2019). *Approche d’ingénierie systèmes pour l’intégration de la fabricabilité dans le processus de conception des systèmes mécatroniques*. PhD thesis, Université Paris-Saclay; Université de Tunis. (Not cited.)
- [Budinger et al., 2014] Budinger, M., Passieux, J.-C., Gogu, C., and Fraj, A. (2014). Scaling-law-based metamodels for the sizing of mechatronic systems. *Mechatronics*, 24(7):775–787. (Cited on page 38.)
- [Cabrera et al., 2010] Cabrera, A. A., Foeken, M., Tekin, O., Woestenenk, K., Erden, M., De Schutter, B., Van Tooren, M., Babuška, R., van Houten, F. J., and Tomiyama, T. (2010). Towards automation of control software: A review of challenges in mechatronic design. *Mechatronics*, 20(8):876–886. (Cited on page 37.)
- [Cao and Xiao, 2007] Cao, P.-B. and Xiao, R.-B. (2007). Assembly planning using a novel immune approach. *The International Journal of Advanced Manufacturing Technology*, 31(7-8):770–782. (Cited on page 27.)
- [Chakrabarty and Wolter, 1997] Chakrabarty, S. and Wolter, J. (1997). A structure-oriented approach to assembly sequence planning. *IEEE Transactions on Robotics and Automation*, 13(1):14–29. (Cited on page 26.)
- [Chapman and Pinfold, 2001] Chapman, C. B. and Pinfold, M. (2001). The application of a knowledge based engineering approach to the rapid design and analysis of an automotive structure. *Advances in engineering software*, 32(12):903–912. (Cited on page 28.)
- [Chen, 1992] Chen, C. P. (1992). Design of a real-time and/or assembly scheduler on an optimization neural network. *Journal of Intelligent Manufacturing*, 3(4):251–261. (Cited on page 27.)
- [Chen et al., 2008] Chen, W.-C., Tai, P.-H., Deng, W.-J., and Hsieh, L.-F. (2008). A three-stage integrated approach for assembly sequence planning using neural networks. *Expert Systems with Applications*, 34(3):1777–1786. (Cited on page 27.)
- [Cramer et al., 1994] Cramer, E. J., Dennis, Jr, J. E., Frank, P. D., Lewis, R. M., and Shubin, G. R. (1994). Problem formulation for multidisciplinary optimization. *SIAM Journal on Optimization*, 4(4):754–776. (Cited on page 29.)

- [Dawson and Duffy, 1999] Dawson, J. C. I. P. D. and Duffy, M. (1999). Scheme-builder: computer aided knowledge based design of mechatronic systems. *Assembly Automation*. (Cited on page 39.)
- [De Fazio and Whitney, 1987] De Fazio, T. and Whitney, D. (1987). Simplified generation of all mechanical assembly sequences. *IEEE Journal on Robotics and Automation*, 3(6):640–658. (Cited on page 27.)
- [De Mello and Sanderson, 1991] De Mello, L. H. and Sanderson, A. C. (1991). Representations of mechanical assembly sequences. *IEEE transactions on Robotics and Automation*, 7(2):211–227. (Cited on page 27.)
- [Demoly et al., 2011] Demoly, F., Yan, X.-T., Eynard, B., Rivest, L., and Gomes, S. (2011). An assembly oriented design framework for product structure engineering and assembly sequence planning. *Robotics and Computer-Integrated Manufacturing*, 27(1):33–46. (Cited on page 27.)
- [Desai and Mital, 2003] Desai, A. and Mital, A. (2003). Evaluation of disassemblability to enable design for disassembly in mass production. *International Journal of Industrial Ergonomics*, 32(4):265–281. (Cited on page 27.)
- [Favi and Germani, 2012] Favi, C. and Germani, M. (2012). A method to optimize assemblability of industrial product in early design phase: from product architecture to assembly sequence. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 6(3):155–169. (Cited on page 26.)
- [Gardan and Matta, 2017] Gardan, J. and Matta, N. (2017). Enhancing knowledge management into systems engineering through new models in sysml. *Procedia CIRP*, 60:169–174. (Cited on page 39.)
- [Gasmi et al., 2013] Gasmi, A., Sprague, M., Jonkman, J., and Jones, W. (2013). Numerical stability and accuracy of temporally coupled multi-physics modules in wind turbine cae tools. In *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 203. (Cited on page 23.)
- [Grundstein and Rosenthal-Sabroux, 2004] Grundstein, M. and Rosenthal-Sabroux, C. (2004). Gameth®, a decision support approach to identify and locate potential crucial knowledge. In *Proceedings 5th European Conference on Knowledge Management*, pages 391–402. Citeseer. (Cited on page 34.)
- [Gu et al., 2008] Gu, T., Xu, Z., and Yang, Z. (2008). Symbolic obdd representations for mechanical assembly sequences. *Computer-Aided Design*, 40(4):411–421. (Cited on page 27.)
- [Guizani, 2016] Guizani, A. (2016). *Approche multi-agents pour la conception optimale des systèmes mécatroniques*. PhD thesis, Université Paris-Saclay; Laboratoire de recherche de Mécanique, Modélisation (Cited on pages 91, 92 and 93.)

- [Guo et al., 2007] Guo, J., Wang, P., and Cui, N. (2007). Adaptive ant colony algorithm for on-orbit assembly planning. In *2007 2nd IEEE Conference on Industrial Electronics and Applications*, pages 1590–1593. IEEE. (Cited on page 27.)
- [Hadj et al., 2018] Hadj, R. B., Belhadj, I., Gouta, C., Trigui, M., Aifaoui, N., and Hammadi, M. (2018). An interoperability process between cad system and cae applications based on cad data. *International journal on interactive design and manufacturing (IJIDeM)*, 12(3):1039–1058. (Cited on page 27.)
- [Hales et al., 2015] Hales, J., Tonks, M., Gleicher, F., Spencer, B., Novascone, S., Williamson, R., Pastore, G., and Perez, D. (2015). Advanced multiphysics coupling for lwr fuel performance analysis. *Annals of Nuclear Energy*, 84:98–110. (Cited on page 22.)
- [Hammadi, 2012] Hammadi, M. (2012). *Contribution à l'intégration de la modélisation et la simulation multi-physique pour conception des systèmes mécatroniques*. PhD thesis, École Centrale Paris. (Cited on page 36.)
- [Hause et al., 2006] Hause, M. et al. (2006). The sysml modelling language. In *Fifteenth European Systems Engineering Conference*, volume 9, pages 1–12. (Cited on page 19.)
- [Hehenberger et al., 2010] Hehenberger, P., Egyed, A., and Zeman, K. (2010). Consistency checking of mechatronic design models. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 44113, pages 1141–1148. (Cited on page 38.)
- [Hong and Cho, 1995] Hong, D. and Cho, H. (1995). A neural-network-based computational scheme for generating optimized robotic assembly sequences. *Engineering applications of artificial intelligence*, 8(2):129–145. (Cited on page 27.)
- [Hong and Cho, 1999] Hong, D. and Cho, H. S. (1999). Generation of robotic assembly sequences using a simulated annealing. In *Proceedings 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human and Environment Friendly Robots with High Intelligence and Emotional Quotients (Cat. No. 99CH36289)*, volume 2, pages 1247–1252. IEEE. (Cited on page 27.)
- [Jacques, 1995] Jacques, F. (1995). *Les systèmes multi-agents : vers une intelligence collective / Jacques Ferber, ... ; préface de Jean-François Perrot*. IIA, Informatique, intelligence artificielle. InterEditions, Paris. (Cited on page 89.)
- [Jennings et al., 1998] Jennings, N. R., Sycara, K., and Wooldridge, M. (1998). A roadmap of agent research and development. *Autonomous agents and multi-agent systems*, 1(1):7–38. (Cited on page 89.)
- [Kanai et al., 1996] Kanai, S., Takahashi, H., and Makino, H. (1996). Aspen: computer-aided assembly sequence planning and evaluation system based on predetermined time standard. *CIRP annals*, 45(1):35–39. (Cited on page 26.)

- [Karayel et al., 2004] Karayel, D., Özkan, S. S., and Keleş, R. (2004). General framework for distributed knowledge management in mechatronic systems. *Journal of intelligent manufacturing*, 15(4):511–515. (Cited on page 39.)
- [Keyes et al., 2013] Keyes, D. E., McInnes, L. C., Woodward, C., Gropp, W., Myra, E., Pernice, M., Bell, J., Brown, J., Clo, A., Connors, J., et al. (2013). Multi-physics simulations: Challenges and opportunities. *The International Journal of High Performance Computing Applications*, 27(1):4–83. (Cited on pages 22 and 24.)
- [Kuo and Zhang, 1995] Kuo, T. and Zhang, H.-C. (1995). Design for manufacturability and design for "x": concepts, applications, and perspectives. In *Seventeenth IEEE/CPMT International Electronics Manufacturing Technology Symposium. 'Manufacturing Technologies-Present and Future'*, pages 446–459. IEEE. (Cited on page 25.)
- [Larses and Adamsson, 2004] Larses, O. and Adamsson, N. (2004). Drivers for model based development of mechatronic systems. In *DS 32: Proceedings of DESIGN 2004, the 8th International Design Conference, Dubrovnik, Croatia*. (Cited on page 39.)
- [Leitão and Restivo, 2006] Leitão, P. and Restivo, F. (2006). Adacor: A holonic architecture for agile and adaptive manufacturing control. *Computers in industry*, 57(2):121–130. (Cited on page 51.)
- [Lv and Lu, 2010] Lv, H. and Lu, C. (2010). An assembly sequence planning approach with a discrete particle swarm optimization algorithm. *The international journal of advanced manufacturing technology*, 50(5-8):761–770. (Cited on page 27.)
- [Ma et al., 2019] Ma, J., Chen, S.-L., Teo, C. S., Tay, A., Al Mamun, A., and Tan, K. K. (2019). Parameter space optimization towards integrated mechatronic design for uncertain systems with generalized feedback constraints. *Automatica*, 105:149–158. (Cited on page 38.)
- [Martin, 2020] Martin, J. N. (2020). *Systems Engineering Guidebook*. CRC Press. (Cited on page 17.)
- [Martins and Lambe, 2013] Martins, J. R. and Lambe, A. B. (2013). Multidisciplinary design optimization: a survey of architectures. *AIAA journal*, 51(9):2049–2075. (Cited on page 29.)
- [Mhenni et al., 2014] Mhenni, F., Choley, J.-Y., Penas, O., Plateaux, R., and Hammadi, M. (2014). A sysml-based methodology for mechatronic systems architectural design. *Advanced Engineering Informatics*, 28(3):218–231. (Cited on page 19.)

- [Monticolo et al., 2015] Monticolo, D., Badin, J., Gomes, S., Bonjour, E., and Chamoret, D. (2015). A meta-model for knowledge configuration management to support collaborative engineering. *Computers in Industry*, 66:11–20. (Cited on pages 34 and 36.)
- [Mørkeberg Torry-Smith et al., 2013] Mørkeberg Torry-Smith, J., Qamar, A., Achiche, S., Wikander, J., Henrik Mortensen, N., and During, C. (2013). Challenges in designing mechatronic systems. *Journal of Mechanical Design*, 135(1). (Cited on page 38.)
- [Motavalli and Islam, 1997] Motavalli, S. and Islam, A.-U. (1997). Multi-criteria assembly sequencing. *Computers & industrial engineering*, 32(4):743–751. (Cited on page 27.)
- [Niknam and Ovtcharova, 2013] Niknam, M. and Ovtcharova, J. (2013). Towards higher configuration management maturity. In *IFIP International Conference on Product Lifecycle Management*, pages 396–405. Springer. (Cited on page 33.)
- [Ohashi et al., 2002] Ohashi, T., Iwata, M., Arimoto, S., and Miyakawa, S. (2002). Extended assemblability evaluation method (aem)(extended quantitative assembly producibility evaluation for assembled parts and products). *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 45(2):567–574. (Cited on page 26.)
- [Saad and Chakhar, 2009] Saad, I. and Chakhar, S. (2009). A decision support for identifying crucial knowledge requiring capitalizing operation. *European Journal of operational research*, 195(3):889–904. (Cited on page 34.)
- [Said et al., 2019] Said, S. B., Benhadj, R., Hammadi, M., Trigui, M., and Aifaoui, N. (2019). Mathematic formulation for the generation of combined paths for mounting parts in assembly. *The International Journal of Advanced Manufacturing Technology*, 104(9):4475–4484. (Cited on page 27.)
- [Sanchez and Mahoney, 1996] Sanchez, R. and Mahoney, J. T. (1996). Modularity, flexibility, and knowledge management in product and organization design. *Strategic management journal*, 17(S2):63–76. (Cited on page 39.)
- [Sanderson et al., 1990] Sanderson, A. C., de Mello, L. S. H., and Zhang, H. (1990). Assembly sequence planning. *AI Magazine*, 11(1):62–62. (Cited on page 26.)
- [Santochi and Dini, 1992] Santochi, M. and Dini, G. (1992). Computer-aided planning of assembly operations: the selection of assembly sequences. *Robotics and Computer-Integrated Manufacturing*, 9(6):439–446. (Cited on page 27.)
- [Sinanoğlu and Börklü, 2005] Sinanoğlu, C. and Börklü, H. R. (2005). An assembly sequence-planning system for mechanical parts using neural network. *Assembly Automation*. (Cited on page 27.)

- [Sobieszczanski-Sobieski, 1995] Sobieszczanski-Sobieski, J. (1995). Multidisciplinary design optimization: an emerging new engineering discipline. In *Advances in structural optimization*, pages 483–496. Springer. (Cited on page 29.)
- [Tedford and Martins, 2010] Tedford, N. P. and Martins, J. R. (2010). Benchmarking multidisciplinary design optimization algorithms. *Optimization and Engineering*, 11(1):159–183. (Cited on page 30.)
- [Thomas et al., 2003] Thomas, U., Barrenscheen, M., and Wahl, F. M. (2003). Efficient assembly sequence planning using stereographical projections of c-space obstacles. In *Proceedings of the IEEE International Symposium on Assembly and Task Planning, 2003.*, pages 96–102. IEEE. (Cited on page 26.)
- [Thramboulidis, 2013] Thramboulidis, K. (2013). Overcoming mechatronic design challenges: the 3+ 1 sysml-view model. *Computing Science and Technology International Journal*, 1(1):6–14. (Cited on page 38.)
- [Tseng et al., 2011] Tseng, Y.-J., Yu, F.-Y., and Huang, F.-Y. (2011). A green assembly sequence planning model with a closed-loop assembly and disassembly sequence planning using a particle swarm optimization method. *The International Journal of Advanced Manufacturing Technology*, 57(9):1183–1197. (Cited on page 27.)
- [van Amerongen, 2003] van Amerongen, J. (2003). Mechatronic design. *Mechatronics*, 13(10):1045–1066. (Cited on page 37.)
- [Vanegas et al., 2018] Vanegas, P., Peeters, J. R., Cattrysse, D., Tecchio, P., Ardente, F., Mathieux, F., Dewulf, W., and Dufflou, J. R. (2018). Ease of disassembly of products to support circular economy strategies. *Resources, Conservation and Recycling*, 135:323–334. (Cited on page 27.)
- [Wang et al., 2005] Wang, J., Liu, J., and Zhong, Y. (2005). A novel ant colony algorithm for assembly sequence planning. *The international journal of advanced manufacturing technology*, 25(11-12):1137–1143. (Cited on page 27.)
- [Wang and Liu, 2010] Wang, Y. and Liu, J. (2010). Chaotic particle swarm optimization for assembly sequence planning. *Robotics and Computer-Integrated Manufacturing*, 26(2):212–222. (Cited on page 27.)
- [Wooldridge and Jennings, 1999] Wooldridge, M. and Jennings, N. R. (1999). The cooperative problem-solving process. *Journal of Logic and computation*, 9(4):563–592. (Cited on page 89.)
- [Yang et al., 2021] Yang, P., Lu, J., Feng, L., Wu, S., Wang, G., and Kiritsis, D. (2021). A knowledge management approach supporting model-based systems engineering. In *Trends and Applications in Information Systems and Technologies: Volume 2 9*, pages 581–590. Springer. (Cited on page 39.)

- [Zhang et al., 2014] Zhang, H., Liu, H., and Li, L. (2014). Research on a kind of assembly sequence planning based on immune algorithm and particle swarm optimization algorithm. *The International Journal of Advanced Manufacturing Technology*, 71(5):795–808. (Cited on page [27](#).)

Research project and perspectives: Collaborative multidisciplinary design of Cyber Physical Systems

Contents

2.1	Introduction	107
2.2	New industrial challenges	107
2.3	New research challenges	108
2.4	New research axis	110
2.4.1	Research axis 1: Digital twin development for cyber physical systems	110
2.4.2	Research axis 2: Interdisciplinary collaborative design of cyber physical (production) systems	113
2.5	Conclusion	114
	References	116

2.1 Introduction

In this chapter, I start by presenting the new industrial and research challenges that led us to define our research project for the more or less long term. The two research axis: digital twin development for cyber physical systems and interdisciplinary collaborative design of cyber physical (production) systems will therefore be detailed.

2.2 New industrial challenges

For a long time, the industrial sector has focused on productivity issues, particularly through the triptych of costs, quality and deadlines. The digitization of processes, the demand for the right level of customer experience, the increase in competitive pressure and the development of technologies are forcing industrial groups to rethink their production processes and to change their paradigm.

The factory of the future brings digital technologies closer to the industrial world. The industrial process of tomorrow will be built on new technologies and innovation. The production tool is transforming, it communicates continuously through the production and supply chain. Industry 4.0 is becoming a true digital platform. The development of Internet of Things (IoT), Artificial Intelligence (AI), Big Data and the Cloud are its main tools. Therefore, the challenge for manufacturers is no longer so much to produce in large quantities, but to be able to adapt their production chain to the needs of their consumers in real time and in a flexible manner.

On the other hand, manufacturing and transportation of goods will see a remarkable growth in the use of autonomous vehicles in the future, whether they are Automated Guided Vehicles (AGVs) or Autonomous Aerial Vehicles/Unmanned Aerial Vehicles (AAV/UAVs). Indeed, these autonomous vehicles (AVs) excel in applications with repetitive movement of materials over a distance or with a process where material tracking is important [Caruntu et al., 2020].

Last but not least, the advent of Industry 4.0 will not happen without taking into account the ecological transition. Reducing the environmental footprint is already a major challenge for industrial groups.

Faced with these industrial upheavals, we are aware that our research must include new methods and tools in the design process of complex systems. New collaboration strategies between the designers need also to be implemented, so that product development companies are able to exploit the new I4.0 technology, to remain competitive and attractive to increase their offer.

2.3 New research challenges

With the advent of the Industry 4.0 concept, customers seek for highly personalized products which requires including customers with design team around a collaboration platform integrating new I4.0 technologies in the product design phase to meet. It also involves the development of new collaboration strategies allowing better use of data, information and knowledge.

The advent of the Industry 4.0 concept has also brought changes to the simulation and modeling paradigm. The new simulation and modeling paradigm evolves today to the concept of “Digital Twin”. This concept extends the use of modeling & simulation to all phases of the product lifecycle, where products are first developed and tested in detail in a virtual environment, and subsequent phases use the information generated and collected by the previous product phases of the life cycle [Rodič, 2017].

The way humans interact with engineered systems is also transforming, just as the internet has transformed the way people interact with information. Indeed with the integration of computation, networking and physical processes, engineered systems are transforming to Cyber Physical System (CPS). CPSs are a way to encompass the large spectrum of technologies including robotics, IoT, embedded systems, and so on. The Autonomous driving and smart traffic, emergency response, environmental monitoring, electric energy grids, space missions, medical applications, and so on are CPS systems that share the challenge of integrating artificial intelligence and that require new design methodologies. Being composed of many heterogeneous elements, CPSs require complex models to define each sub-system and its behavior. Current design tools need to be upgraded to consider the interactions between the various sub-systems, their interfaces, and abstractions.

The application of the CPS to industrial manufacturing leads to a specialization of them referred as Cyber Physical Production System (CPPS). CPS and CPPS should be able to address interoperability issues, since one of their intrinsic requirement is the capability to interface and cooperate with other systems.

The design and control challenges of CPSs are harder, since individual component autonomy must be glued by explicitly addressing interdependence and coordination, interoperability, distributed control and emergence of behaviors. A strong demand for address today to the implementation of swarm intelligence in CPSs which play an important role when there is not enough information to solve the design problem in a centralized way, when there are time constraints which do not allow to find an analytical solution, and when the operation needs to be performed in a dynamically changing environment. Swarm Intelligence (SI) is a popular multi-agent framework that has been originally inspired by swarm behaviors observed in natural systems, such as ant and bee colonies.

Swarm intelligence is based on the principle of thinking globally and acting locally [Dias-Ferreira et al., 2018]. For instance, swarming uses the principle of collective intelligence for solving the problem of traveling in formations of AVs. The vehicles can be considered as simple particles that cooperate while updating their velocities. Thus, bio-inspired techniques to control a fleet of AVs could be applied to accommodate their operations in unexplored and uncertain environments. The tasks of each entity are therefore derived from emergent cooperative biological systems. Consequently, this allows considering decentralized, self-organized formations, in which each participant can have a specific behavior [Dias-Ferreira et al., 2018].

This introduction leads us to formulate the scientific issues on which we are led to make contributions in the coming years. In the next section, we will detail our new research topics to consider the following three scientific issues:

- How to make classical modeling & simulation techniques of complex systems evolve to the digital twin concept?
- how to make mechatronic design methodologies evolve towards the design of complex systems becoming more and more into **CPSs**?
- and, how to develop new collaborative design platforms integrating I4.0 technologies and allowing both designers and clients to collaborate in an efficient manner.

2.4 New research axis

2.4.1 Research axis 1: Digital twin development for cyber physical systems

The digital twin aims to produce a virtual replica of a physical asset (machine, system, product) in order to obtain visualization, simulation / prediction and monitoring functionalities. In the context of the Factory, the Digital Twin is an approach making it possible to couple the physical world with digital tools, and this throughout the life cycle of the system concerned and at all levels (strategic, tactical, operational, real time). The concept is today at the heart of many attentions and many questions, research, organizational and technological are raised. According to their typology, we can consider the development of digital twins for the two following research activities:

- Multiphysics simulation of cyber-physical systems
- Design of cyber-physical production systems

2.4.1.1 Multiphysics simulation of cyber-physical systems

The fields such as Aeronautics, Space, Land Transport, Defense, Security and Digital Identity integrate electronic modules (digital cards, analogue cards, power supply cards, mixed digital/analogue cards, ...) with software programs (Signal Processing, Specific Wave-forms, Low Level Software, Secure OS, ...), all this supported and protected by mechanical structures (assembly, materials of various types and shapes, ...), for the development of their complex engineered systems considered today as **CPSs**. This imposes a set of design constraints defined according to the specifications of the **CPS** to be developed.

In order to meet these constraints, Multiphysics Digital Twin (**MPDT**) is considered as an essential tool that helps at the same time: in the rapid definition of the architectures, the detailed exploration of the **CPS** equipment's in their environments and the reduction of the risks linked to a bad anticipation of certain physical or functional phenomena. **MPDT** also limits the costs by allowing

the design to converge more quickly and reducing the number of physical prototypes.

To do so, a **MPDT** is currently being developed as a part of Grégoire GALISSON's Thesis [Th9], which has started in May 2021. It is expected that the digital twin will be able to handle both the simulation management and the simulation modeling activities.

The simulation management helps in taking control of simulation data and processes to avoid common problems, such as analyzes performed on stale data, poor visibility of simulation results, and their coming too late to affect performance. Thus, from the specifications of the equipment to be designed and the analysis needs, the **MPDT** should help the System Modeling Architect (SMA) of the product in developing the specifications for the simulation model (inputs, outputs, functional and environmental fields of application, physics involved, exchange formats between physics, temporal aspects and tools). The **MPDT** should also help the system modeling architect in setting up the simulation workflow and the coordination between the different stakeholders involved in the development and use of the simulation model, including verification and validation phases of the digital twin.

The simulation modeling will take into account the aspects of couplings between physics:

- the weak or strong coupling between 3D simulation tools and physics, which often does not offer the possibility of going beyond two or three physics,
- the coupling of reduced models within a system simulation which opens up the possibility of a system multiphysics platform.

To do this, the thesis is stained with defining an evolving theoretical, methodological and application framework allowing the design of the **MPDT**, as well as its implementation and its coupling with experimental devices (prototypes of the real system). This framework is required to allow the digital twin to evolve throughout the product design process and its cycle of use.

The use of model reduction techniques to develop system simulation models is not recent, however classical model reduction approaches are essentially based on a global application of modal decomposition techniques. The major drawback of these techniques is that these methods often lead to inaccuracies and instabilities in the solution caused, among other, by certain truncated modes which may contain a significant amount of information and system energy.

The contribution targeted in this thesis is to develop a systematic approach for model order reduction using adaptive partitioning and clustering techniques to customize partitions with properly chosen widths. Each partition characterizes a specific stage in the dynamics and evolution of the system. Partitioning can be done

on physical domains, state space, parameter space, and time domain. During the use phase of the MPDT, an automatic calibration of the DT configuration parameters will be required. For this, we propose to use a method assisted by surrogates models and space mapping to align the digital twin with the data measured from the real system. The trained surrogates should be re-optimized by updating the system configuration in order to maximize the consistency between the MPDT predictions and the measurements of the physical system sensors.

2.4.1.2 Design of cyber-physical (production) systems

Personalized products and production require new development techniques to overcome the main hurdles of the customization of complex engineered systems. Typically, production processes for various complex product groups struggle to respond to the dynamic situations that arise from complex production processes. In this sense, Digital Twin already presents promising avenues to overcome these challenges.

CPS and CPPS are core concepts of the fourth industrial revolution. They are physical and engineering systems that can overcome the limitations of the personalized products and production process, as they are designed to improve the performance of products and manufacturing systems [Monostori et al., 2016]. In the manufacturing sector, a digital twin can be defined as a virtual plant that reflects the structure of a real plant and synchronizes information and functions related to the plant design, operation, and production [Park et al., 2020]. However, several challenges are still hindering an effective development of digital twins for CPS and CPPS. Indeed, the collection of accurate customer requirements and the reduction of the time gap between product development stakeholders (customer, designer, manufacturer, etc) are main challenges to overcome.

For this, we propose in this research axis, to develop new methodologies allowing the effective integration of digital twins for CPS and CPPS design. For this, several challenges need to be considered, among them:

- the effective integration of CPS or CPPS customers' requirements in the Digital Twin.
- the evolution of classical 3D CAD models towards a 3D digital twin integrating knowledge and artificial intelligence
- the digitalisation of manufacturing based on the digital twin

These research activities are already carried out in the framework of a collaborative project that I am piloting (Franco-Tunisian Hubert Curien Partnership (PHC) program - UTIQUE) with the mechanical engineering laboratory LGM of the National Engineering School of Monastir (ENIM) in Tunisia. This project deals with the digitization of manufacturing systems using I4.0 technologies, where

the digital twin will play a main role. Indeed, the Digital Twin is considered as a relevant solution for digitizing manufacturing. In this context, our aim is to develop new approaches allowing the evolution of traditional CAD models towards a form of Digital Twin representing manufacturing units. The digital twin will be able to learn by itself from data transmitted with different formats from several sources: sensors of the physical system, relevant knowledge of the industrial field, other similar machines and the environment in which the physical system is making part.

As second research activity, we planned to work on the development of new design methodologies for the assemblability assessment of smart products and CPSs. Indeed, the constantly growing needs of customized products impose use of sensors, which provide feedback information on CPS use, behaviors and states in real time. The few works done to date on assembly analysis methods do not take this constraint into account. It would therefore be a question to develop new methods of analysis of ASP/DSP for CPSs.

As a third topic related to the design of CPS/CPPS, we started within the framework of the thesis of Khalil ALOUI [Th8], to work on the development of decentralized control systems based on swarm robotics and bioinspiration. The activities for this work concern the continuity of the design process starting from the description of swarm behaviors, such as swarm formation, to the functional design of swarm element, the structural and architectural design of robots, the development of bioinspired control algorithms and the implementation of programs. SysML language is already being used in this work for the systems engineering phase of the CPS/CPPS, and Robot Operating System (Robot Operating System (ROS)) is used for the verification and code implementation. The interoperability between SysML and ROS is one of the challenges to overcome. The main contributions of this work will concern methods and tools to integrate swarm intelligence for the design and control of CPS/CPPS.

2.4.2 Research axis 2: Interdisciplinary collaborative design of cyber physical (production) systems

The work we have started on the collaborative design of mechatronic systems has allowed us to acquire skills in the field of knowledge management (KM) and the use of artificial intelligence (AI) through multi-agent systems. This opens the horizons for us to combine these skills to develop new methods of collaboration for the design of CPS and CPPS. Indeed, the extension of ICEs to include design knowledge formulated as an optimization sub-problem with the solution to the sub-problem presented as an array of discrete solutions can help in reducing the development time, taking into account the new design constraints.

On the other hand, if we examine more deeply into modern collaborative design practices, we can notice that the internet of things and connected objects, big data analytics, smart systems, etc. are increasingly being integrated into knowledge

management processes. This stimulated the development of integrated software platforms to optimize knowledge management strategies. In this context, knowledge management practices in the design process require the technology capabilities that support Industry 4.0, such as digital enhancements and the implementation of CPS, which can facilitate the process of managing knowledge. A such approach will take into account the technological challenges that collaborative design is dealing with in an Industry 4.0 context. Therefore, these challenges involve diverse stakeholders' skills and creative capabilities.

The issues to be considered in this case could be:

- How can we develop new knowledge using IoT and connected objects that improve decision-making in collaborative design?
- How can big data help in automating the design collaboration process and provide design insights based on data analysis?
- How the cloud can provide DFX data and information on production optimization opportunities?
- How artificial intelligence tools such as machine learning algorithms, natural language processing, text processing, etc., coupled with big data and advanced algorithms can be helpful in using unstructured data sources to help designers in codifying knowledge and address the missing links in their knowledge flow?

Therefore, by considering the Industry 4.0 enablers in the collaborative design, we believe that we can develop self-learning design and coordinating agents, using deep learning and reinforcement learning techniques, which allows both the automatic capitalization of knowledge and the effective assistance of virtual design agents to real design agents.

Thus, we intend to develop this research axis for the collaborative design of CPS and CPPS.

2.5 Conclusion

Our research project for the next few years is therefore based on the integration of Industry 4.0 technologies in the collaborative design process of more intelligent, communicating and collaborative systems, namely CP(P)Ss.

In this chapter I have presented the two axes of research that we wish to develop in the near future: Digital twin development for cyber physical systems and Interdisciplinary collaborative design of cyber physical (production) systems. This goes in the same direction of a continuity of the two main axes of research: integrated design and collaborative design, but considering new challenges on the integration of I4.0, and later I5.0, technologies such as IoT, big data analytics, cloud computing, digital twins and artificial intelligence.

The objective is always to formalize modeling and design frameworks in order to

master the growing complexity of products and production processes, and to guarantee that the design requirements are respected.

References

- [Caruntu et al., 2020] Caruntu, C. F., Pascal, C. M., Maxim, A., and Pauca, O. (2020). Bio-inspired coordination and control of autonomous vehicles in future manufacturing and goods transportation. *IFAC-PapersOnLine*, 53(2):10861–10866. (Cited on page 108.)
- [Dias-Ferreira et al., 2018] Dias-Ferreira, J., Ribeiro, L., Akillioglu, H., Neves, P., and Onori, M. (2018). Biosoarm: a bio-inspired self-organising architecture for manufacturing cyber-physical shopfloors. *Journal of Intelligent Manufacturing*, 29(7):1659–1682. (Cited on page 109.)
- [Monostori et al., 2016] Monostori, L., Kádár, B., Bauernhansl, T., Kondoh, S., Kumara, S., Reinhart, G., Sauer, O., Schuh, G., Sihn, W., and Ueda, K. (2016). Cyber-physical systems in manufacturing. *Cirp Annals*, 65(2):621–641. (Cited on page 112.)
- [Park et al., 2020] Park, K. T., Lee, J., Kim, H.-J., and Do Noh, S. (2020). Digital twin-based cyber physical production system architectural framework for personalized production. *The International Journal of Advanced Manufacturing Technology*, 106(5):1787–1810. (Cited on page 112.)
- [Rodič, 2017] Rodič, B. (2017). Industry 4.0 and the new simulation modelling paradigm. *Organizacija*, 50(3). (Cited on page 108.)

General conclusion

The theme of collaborative multidisciplinary design of complex systems is still relevant today.

The research works of the last years have been focused on the methods of modeling and integrated collaborative design of mechatronic systems. In recent years and in the future this theme opens on the connected, communicating and collaborative systems, known as CPS and CPPS.

Thus to master the ever-increasing complexity of products whose requirements are evolving in an industrial environment itself in evolution, the methods and tools for modeling, verifying and validating the design of complex systems are also constrained to evolve to meet the new challenges of product and process development.

This research theme is very open and presents several major technological and methodological challenges that will need to be resolved in the years to come, notably regarding the integration of I4.0 technologies: IoT, big data, cloud computing, digital twins and artificial intelligence.

Collaborative Multidisciplinary Design of Complex Systems

Abstract: In this HDR manuscript, I have presented a review on our research themes concerning the formalization of modeling and design frameworks in order to master the increasing complexity of mechatronics and CPS products, and production processes (CPPS).

Therefore, in the first chapter I presented a synthesis on the work done to develop a formal framework of modeling and design allowing collaborative design and management of the knowledge related to the activities in systems engineering, mechatronic design, multiphysics simulation, DFX and MDAO, with applications to mechatronic products.

In the second chapter, I presented our research project which aims at integrating the industry 4.0 enablers (IoT, big data, cloud computing, digital twins and artificial intelligence) for the integrated and collaborative design process of CP(P)Ss. Therefore I detailed the two research axis: Digital twin development for cyber physical systems and Interdisciplinary collaborative design of cyber physical (production) systems. The objective is always mastering the growing complexity of such products and production processes, and to guarantee that the design requirements are respected.

Keywords: Complex Engineered Systems, Mechatronics, CPS, CPPS, MBSE, Knowledge Management, Collaborative Design, Digital Twin, Artificial Intelligence.
