
CoSim20: un environnement pour la simulation collaborative et distribuée

Julien Deantoni*¹

¹Laboratoire d'Informatique, Signaux, et Systèmes de Sophia Antipolis (I3S) – CNRS : UMR7271, Université Côte d'Azur (UCA), Polytech Nice-Sophia, Inria Sophia Antipolis - Méditerranée, GEMOC Initiative – 2004 rte des Lucioles (Lagrange L-041), BP93, F-06902 Sophia Antipolis Cedex, France

Résumé

Les systèmes cyber-physiques sont des systèmes d'ingénierie complexe dans lesquels les parties computationnelles communiquent entre elles et avec les parties physiques décrivant l'environnement. Pour apprivoiser la complexité croissante de ces systèmes, ils sont généralement décomposés en différentes parties qui sont modélisées par différents experts, éventuellement issus de différentes organisations. Ces experts utilisent un langage adapté à leur problème, à la fois syntaxiquement et sémantiquement. À ce stade, d'une part on obtient des modèles exécutables des parties computationnelles et des modèles exécutables des parties physiques et, d'autre part, les modèles exécutables ne doivent pas violer les propriétés intellectuelles lorsqu'ils sont partagés et sont donc généralement partagés en tant qu'unités de simulation en boîte noire. Cependant, pour comprendre le comportement émergent de l'ensemble du système, il est nécessaire de faire une simulation collaborative ; où les unités de simulation de différentes disciplines sont coordonnées pour échanger des données aux moments opportuns.

Les problèmes avec les approches existantes sont multiples. Premièrement, les unités de simulation exposées sous forme de boîtes noires ne permettent pas de prendre en compte les spécificités sémantiques de chaque modèle. Deuxièmement, les co-simulations sont aujourd'hui principalement basées sur une interface de programmation dirigée par le temps, et il a été montré que de telles interfaces introduisaient, lorsqu'elles sont appliquées à des systèmes cyber-physiques, des "retards artificiels" dus à la co-simulation elle-même. De tels retards impliquent une mauvaise précision qui peut invalider les résultats de la co-simulation. De plus, réduire ces délais de co-simulation implique de mauvaises performances globales de co-simulation. Troisièmement, la définition d'un cadre de co-simulation précis et performant peut être complexe d'un point de vue algorithmique, de sorte qu'un support est nécessaire pour augmenter le niveau d'abstraction lors de la définition de la coordination.

La présentation, **basée sur les travaux de thèse de Giovanni Liboni** tendra à montrer qu'il est possible 1) de fournir une interface de coordination de niveau modèle qui englobe à la fois les modèles cyber et les modèles physiques d'un système cyber-physique et 2) de définir un langage dédié à la coordination de tels modèles. Sur la base de ces artefacts et d'une interface de co-simulation originale, il est possible de générer automatiquement une infrastructure distribuée précise et performante pour la co-simulation.

Cette proposition est soutenue par la mise en œuvre de deux langages dédiés. Un pour la

*Intervenant

définition de l'interface de coordination du modèle et un pour la définition de la spécification de la coordination. Un prototype d'API respectueux de la sémantique comportementale des modèles, qui est introduite comme une généralisation des API standard existantes étaye la proposition. Enfin, un compilateur a été défini pour que les coordinations définies avec les langages proposés puissent être utilisées pour générer automatiquement un environnement de co-simulation distribué. Les différentes propositions sont appliquées sur une étude de cas où les avantages de l'approche sont clairement illustrés.