

D.I.S.Co. :
UN NOUVEL ENVIRONNEMENT DE SIMULATION ORIENTE OBJET

A. Sargousse¹

J.M. Le Lann¹

X. Joulia¹

L. Jourda²

1 : Laboratoire de Génie Chimique (L.G.C., U.M.R. C.N.R.S. 5503), Equipe Analyse Fonctionnelle des Procédés, 18, Chemin de la Loge, 31078 Toulouse Cedex 04 (France)

2 : ProSim S.A., 132, Route d'Espagne, 31100 Toulouse (France)

Auteur responsable de la correspondance :

A. Sargousse
L.G.C. C.N.R.S. U.M.R. 5503
18, Chemin de la Loge
31078 Toulouse Cedex 04 (France)
Tel : +33 (0) 562 252 355
Fax : +33 (0) 562 252 318
Email : Alain.Sargousse@ensigct.fr

Mots-Clefs : Solveurs Numériques, U.M.L. (Unified Modeling Language), Approche Orientée objet, Simulation Dynamique

I. INTRODUCTION :

L'établissement de modèles mathématiques rigoureux en vue de la simulation dynamique de procédés chimiques conduit généralement à des systèmes mixtes d'équations différentielles ordinaires (E.D.O.) ou partielles (E.D.P.) (pour les bilans matières, d'énergie et de quantité de mouvement) et d'équations non linéaires algébriques (E.A.N.L.) (pour les équations phénoménologiques telles que les cinétiques, les équilibres physiques et l'hydrodynamique) ; Ces systèmes d'équations sont appelés équations différentielles et algébriques (E.D.A.) ou équations différentielles partielles et algébriques (E.D.P.A.).

Ces systèmes présentent habituellement les caractéristiques suivantes :

- Systèmes de grande taille à structure creuse
- Forte non linéarité
- Large spectre de constante de temps (problème de « stiffness »)
- Systèmes sous contraintes événementielles (occurrence d'événements d'état ou temporels)

De ce constat, il apparaît que toute simulation rigoureuse de procédés chimiques nécessite l'utilisation d'un environnement numérique performant à même de traiter les caractéristiques de ces systèmes. Cet environnement doit être le plus général possible afin d'être en mesure de simuler le maximum de modèles de génie chimique. Le but ultime étant de déconnecter définitivement modélisation et résolution de telle sorte que l'ingénieur de procédés puisse se consacrer exclusivement à la formulation de son modèle.

II. L'ENVIRONNEMENT NUMERIQUE :

Le développement d'outils généraux de simulation de procédés chimiques peut rapidement conduire à un travail pénible lorsque des techniques de programmation procédurale sont utilisées ; une approche orientée objet peut alors être d'une aide certaine pour assurer la réutilisabilité et l'extensibilité des codes. C'est pourquoi, l'environnement numérique décrit dans cet article a été conceptualisé en U.M.L. (*Unified Modeling Language*) (cf. Figure I) et implémenté en langage C++.

Pour ce qui concerne la modélisation, à partir des concepts développés par L. Jourda pour la simulation en régime permanent [1], trois classes de bases ont été développées : La classe VARIABLE, la classe EQUATION et la classe MODELE. La VARIABLE est un objet permettant de stocker une valeur ainsi que toutes ses dérivées éventuelles. Les dérivées d'une VARIABLE sont également des VARIABLES. Une VARIABLE peut représenter une inconnue ou un paramètre, ce choix se faisant uniquement à la résolution ouvrant la porte à la simulation dite inverse. L'EQUATION est un objet utilisant un certain nombre de VARIABLES et fournissant des services permettant de calculer le résidu ainsi que les dérivées de l'EQUATION par rapport à toutes ses VARIABLES. Enfin, l'objet MODELE est composé d'un certain nombre d'EQUATIONS et de VARIABLES. Les VARIABLES du MODELE n'étant que l'ensemble de toutes les VARIABLES utilisées par les EQUATIONS (cet ensemble assurant l'unicité de chaque VARIABLE). Le MODELE présente la capacité de se fusionner avec d'autres MODELES par fusion des ensembles d'EQUATIONS et de VARIABLES. Cette caractéristique est particulièrement intéressante pour effectuer une simulation globale d'un flowsheet à partir des MODELES élémentaires des modules de génie chimique le composant. De même, l'utilisation d'équations de connexion entre modules devient caduque puisque deux MODELES peuvent partager les mêmes VARIABLES. A l'heure actuelle, la classe MODELE permet de décrire n'importe quel système E.D.O., E.A.N.L. ou E.D.A. La généricité de l'écriture d'un modèle permet de s'affranchir du type de simulation souhaitée : le même MODELE pouvant être indifféremment simulé en régime permanent ou en dynamique. Des classes VARIABLE et EQUATION vectorielles ont également été développés afin de pouvoir efficacement représenter, par exemple, des compositions chimiques et des bilans matière partiels.

Pour la partie résolution, des classes SOLVEURS ont été créés :

- un SOLVEUR de systèmes linéaires résultant de l'encapsulation du solveur procédural M.A.38.
- un SOLVEUR d'équations non linéaire de type Newton-Raphson utilisant le SOLVEUR précédent.
- Enfin, un SOLVEUR d'équations différentielles et algébriques général

Ce SOLVEUR résulte de l'encapsulation d'un intégrateur procédural développé au sein de notre équipe. Issu originellement de la plate-forme L.S.O.D.I. de Hindmarsh [2] et utilisant la méthode de Gear [3], cet intégrateur a été fortement amélioré afin de prendre en compte le calcul des conditions initiales cohérentes, la détection des événements temporels ou d'état ainsi que le traitement des systèmes creux. L'encapsulation de cet intégrateur permet de conserver un noyau numérique performant.

L'ensemble de ces objets SOLVEURS offrent des interfaces claires et cohérentes permettant une utilisation beaucoup plus simple que leur alter ego procédural.

Enfin, pour faire communiquer les objets de la partie modélisation avec les objets de la partie résolution, une classe appelée MAP a été créée. Cet objet MAP permet d'adapter les MODELES au SOLVEUR utilisé. Ainsi, les VARIABLES paramètres sont écartées pour ne conserver que les inconnues. L'objet MAP permet également d'exploiter la structure creuse des MODELES par création automatique de l'opérateur dynamique. La simulation continue d'un modèle passe par l'association d'un objet MODELE, d'un objet SOLVEUR et d'un objet MAP.

Pour écrire un modèle dans cet environnement, le modélisateur doit créer ses propres objets VARIABLES, EQUATIONS et MODELES en héritant des classes de base décrites plus haut. L'utilisation du langage C++ pouvant être rebutante pour écrire un modèle, un **Générateur de Modèles** a été développé. Cet outil, à travers une interface graphique, permet de décrire aisément un modèle à travers les expressions des équations le composant. Le **Générateur de Modèle** se charge alors de générer le code C++ décrivant ce modèle au travers des classes utilisables dans l'environnement numérique. Cet outil s'avère fort utile aussi bien en conception de modèles qu'en analyse. Il accélère de façon notable le temps de développement.

L'ensemble des classes décrites ci-dessus permet d'effectuer d'ores et déjà des simulations rigoureuses continues or, la plupart des procédés chimiques actuels présentent à des degrés divers des caractéristiques discrètes (fonctionnements cycliques ou semi-continus, gestion des ressources etc...). Ce type de simulation, mêlant étroitement simulation continue et simulation discrète, est généralement appelé hybride [4]. Pour pouvoir

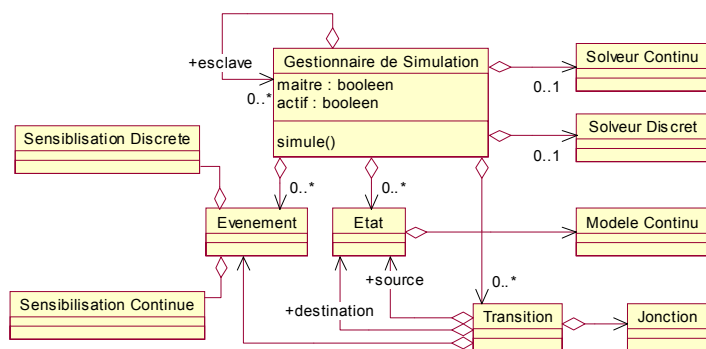


Figure 1 - Diagramme U.M.L. simplifié

tenir compte de façon significative de ce comportement mixte, de nouvelles classes ont été créées, et plus particulièrement la classe GESTIONNAIRE DE SIMULATION comprenant un SOLVEUR Continu (un SOLVEUR E.D.A. en l'occurrence) et un SOLVEUR Discret (dans notre cas, une machine à états). La communication entre ces deux SOLVEURS est assurée par le GESTIONNAIRE DE SIMULATION. Le SOLVEUR Discret se charge de créer le MODELE à simuler dans l'état courant ainsi que d'y associer les EVENEMENTS à surveiller. Le SOLVEUR Continu se charge d'intégrer le MODELE courant et

de détecter l'occurrence d'un EVENEMENT. A chaque module de génie chimique est associé un GESTIONNAIRE DE SIMULATION. Sachant que d'un point de vue topologique un module peut lui-même être composé d'autres modules (un module Colonne est composé de modules Plateau), on retrouve au niveau du GESTIONNAIRE DE SIMULATION le même type de décomposition : un GESTIONNAIRE DE SIMULATION peut lui-même être composé d'un certain nombre de GESTIONNAIRES DE SIMULATION dit *Esclave*. Dans le cadre d'une simulation globale, après chaque événements, le GESTIONNAIRE dit *Maître* (c'est à dire, responsable de l'intégration et généralement appartenant au module Flowsheet) se charge de fusionner l'ensemble des MODELES courants de ses GESTIONNAIRES *Esclave* actifs par requêtes récursives (un GESTIONNAIRE inactif représentant un module non pris en compte dans l'état courant (appareil éteint, etc...)). Cette architecture ouverte permet la modélisation de la plupart des procédés chimiques hybrides et pourrait, à l'avenir, permettre d'envisager la simulation dynamique modulaire [5].

III. CONCLUSION

Un environnement numérique D.I.S.Co. (**Do Integrate by Software Components**) a été développé dans le cadre de l'élaboration d'un simulateur dynamique général. Cet environnement permet de déconnecter modélisation et résolution. Il propose à la fois des méthodes de résolution performantes au niveau continu et une architecture permettant la simulation de systèmes discrets. Afin d'être plus général, il devra néanmoins être complété par des classes permettant la modélisation et la résolution de systèmes E.D.P.A. très courants en génie chimique (réacteurs tubulaires, colonnes à garnissage etc...).

IV. BIBLIOGRAPHIE

- [1] L. Jourda, « Composants Logiciels Orientés-Objet pour la Modélisation et la Simulation des Procédés Chimiques », Thèse I.N.P.T. (Toulouse) (1996)
- [2] A.C. Hindmarsh, « ODEPACK, A Systematic Collection of ODE Solvers », Scientific Computing, eds R.S. Stepleman et al., North-Holland (Amsterdam) (1983)
- [3] C.W. Gear, « Simultaneous Numerical Solution of Differential Equations », IEE Trans. Circ. Theory, CT-18, 1031-1035 (1971)
- [4] R. Champagnat, « Supervision de Systèmes Discontinus : Définition d'un Modèle Hybride et Pilotage en Temps Réel », Thèse U.P.S., L.A.A.S. (Toulouse) (1998)
- [5] F. Laganier, « Simulation Dynamique de Procédés - Méthodes Itératives Dynamiques pour la Résolution de Systèmes Algèbro-Différentiels », Thèse I.N.P.T. (Toulouse) (1993)