

MODELISATION ET SIMULATION

DE MODELES MIN-MAX DEVS HIERARCHIQUES

N. Giambiasi, M. Hamri, C. Frydman

LSIS

UMR CNRS 6168

Université Paul Cézanne Aix-Marseille III

{norbert.giambiasi, amine.hamri, claudia.frydman}@lsis.org

RESUME : Nous avons proposé, dans un précédent article, le formalisme Min-Max DEVS pour la spécification et la simulation de modèles comportementaux pour lesquels la durée de vie des états ne peut être déterminée avec précision. Dans le prolongement de ces travaux, nous présentons, ici, un cadre théorique pour la modélisation et la simulation des modèles hiérarchiques Min-Max DEVS. En particulier, nous démontrons la fermeture sous couplage du formalisme ce qui permet, entre autres, la réutilisation des modèles Min-Max DEVS de base dans l'objectif d'effectuer des descriptions hiérarchisées.

Nous présentons, ensuite, un coordinateur hiérarchique qui permet la simulation de modèles Min-Max DEVS couplés. Un exemple complet illustre les concepts de la modélisation et la simulation en Min-Max DEVS.

MOTS-CLES : DEVS, Spécification, Simulation à événements discrets, Modularité et réutilisation.

1 INTRODUCTION

La plupart des formalismes à événements discrets, comme DEVS, permettent de modéliser et simuler des systèmes avec des dates précises d'événements. La date d'un événement de sortie dépend, généralement, de celle d'un événement d'entrée et de retards existant dans le modèle. Dans DEVS, les retards s'expriment par des durées de vie qui sont associées aux états transitoires. Cependant, dans un processus de conception ou d'analyse, la durée des états transitoires est rarement connue avec précision. Par exemple, dans le domaine des circuits, les constructeurs fournissent des valeurs limites et une valeur moyenne du retard de la porte. Ceci a conduit à l'introduction du concept de retards min-max dans les simulateurs de ce domaine (Giambiasi, 1980). Les modèles ainsi obtenus sont alors plus proches de la réalité sans que les durées de simulation s'en trouvent lourdement pénalisées. Dans le prolongement de ces approches, nous avons proposé les Min-Max DEVS et détaillé la simulation d'un modèle atomique de ce type dans (Giambiasi et Ghosh, 2001). Dans ce formalisme, la durée de vie des états est donnée par un intervalle de temps dont les bornes représentent la durée minimale et la durée maximale de l'état transitoire considéré.

Dans le prolongement des travaux présentés dans (Giambiasi et Ghosh, 2001) (Hamri et al., 2006), nous proposons, dans ce papier, un cadre formel pour la description des modèles Min-Max DEVS couplés, cadre indispensable à la construction de modèles hiérarchisés et à leur réutilisation. Après un rappel sur le formalisme Min-Max DEVS et ses fondements, nous démontrons une de ses propriétés de base : la fermeture sous couplage. Ensuite, nous détaillons la structure et les mécanismes de simulation de modèles hiérarchiques. Un

exemple simple illustre les différents concepts de modélisation et de simulation en Min-Max DEVS.

2 RAPPELS

2.1. Le formalisme DEVS

La spécification d'un modèle avec le formalisme DEVS (Zeigler, 1976) (Zeigler, 1984) (Zeigler et al., 2000) consiste à décrire la structure suivante :

$$M = (X, Y, S, \delta_{\text{int}}, \delta_{\text{ext}}, \lambda, D)$$
 avec

- X : est l'ensemble des événements externes d'entrée,
- Y : est l'ensemble des événements externes de sortie,
- S : est l'ensemble des états éventuellement infini,
- $\delta_{\text{int}} : S \rightarrow S$, est la fonction de transition interne causée par l'occurrence des événements internes,
- $\delta_{\text{ext}} : Q \times X \rightarrow S$, est la fonction de transition externe causée par l'occurrence des événements externes d'entrée (avec Q défini ci-après),
- $\lambda : S \rightarrow Y$, est la fonction de sortie,
- $D : S \rightarrow \mathbb{R}^+$, est la fonction de durée de vie d'un état qui représente le temps maximal pendant lequel le modèle peut demeurer dans un état $s \in S$.

L'état est stable (ou passif) si la durée de vie est infinie, sinon il est transitoire (ou actif).

(Zeigler, 1976) a introduit le concept des états totaux Q (*Total States*) d'un système, défini comme suit :

$Q = \{(s, e) \mid s \in S \text{ et } 0 \leq e \leq D(s)\}$ où e est le temps écoulé dans l'état s . Ce concept est fondamental, il permet de spécifier :

- des transitions externes qui sont fonction du temps écoulé dans l'état, et

- dans le cas d'un état transitoire, de définir l'état futur quand la durée de vie de l'état est égale au temps écoulé.

2.1.1. DEVS Modulaires et Couplés

Grâce au concept de ports d'entrée-sortie, il est possible de définir des modèles atomiques modulaires. Cette modularité et le concept de modèles couplés autorisent la construction hiérarchisée de modèles.

La définition de modèles modulaires conduit à introduire un concept d'interface qui est composée de ports d'entrée-sortie par lesquels passent toutes les interactions avec l'environnement. Par suite, un modèle modulaire DEVS est défini par la structure :

$$MA = (X, Y, S, \delta_{int}, \delta_{ext}, \lambda, D) \text{ avec}$$

- $X = \{(p,v) \mid p \in \text{Inports}, v \in X_p\}$: est l'ensemble des ports d'entrée et leurs valeurs,
- $Y = \{(p,v) \mid p \in \text{Outports}, v \in Y_p\}$: est l'ensemble des ports de sortie et leurs valeurs,
- S : est l'ensemble des états séquentiels,
- $\delta_{int} : S \rightarrow S$, est la fonction de transition interne,
- $\delta_{ext} : Q \times X \rightarrow S$, est la fonction de transition externe,
- $\lambda : S \rightarrow Y$, est la fonction de sortie,
- $D : S \rightarrow R^+ \times R^+$, est la fonction de durée de vie, et
- $Q = \{(s,e) \mid s \in S, 0 \leq e \leq D(s)\}$: est l'ensemble des états totaux.

Un modèle DEVS-Couplé est la structure :

$$MC = (X, Y, D, \{M_d \mid d \in D\}, \text{EIC}, \text{EOC}, \text{IC}, \text{select})$$

- X : est l'ensemble des ports d'entrée,
- Y : est l'ensemble des ports de sortie,
- D : est l'ensemble des noms des composants,
- $\{M_d \mid d \in D\}$: est l'ensemble des modèles DEVS décrivant chaque sous-composant M_d .

$M_d = (X_d, Y_d, S, \delta_{int}, \delta_{ext}, \lambda, D)$ est un modèle DEVS avec ports où

$$X_d = \{(p,v) \mid p \in \text{Inports}_d, v \in X_{p_d}\}$$

$$Y_d = \{(p,v) \mid p \in \text{Outports}_d, v \in Y_{p_d}\},$$

- **EIC** : (*External Input Coupling*) est le couplage externe des ports d'entrée vers les ports d'entrée des sous-composants.

$$\text{EIC} \in \{((MC, ip_{MC}), (d, ip_d)) \mid ip_{MC} \in X, d \in D, ip_d \in \text{Inports}_d\},$$

- **EOC** : (*External Output Coupling*) est le couplage externe des ports de sortie des sous-composants à des ports de sortie.

$$\text{EOC} \in \{((d, op_d), (MC, op_{MC})) \mid d \in D, op_d \in \text{Outports}_d, op_{MC} \in Y\}$$

- **IC** : (*Internal Coupling*) est le couplage interne entre des ports de sortie et des ports d'entrée des sous-composants.

$$\text{IC} \in \{((a, op_a), (b, ip_b)) \mid a, b \in D, op_a \in \text{Outports}_a, ip_b \in \text{Inports}_b\}, \text{ et}$$

- **Select** : fonction de priorité entre composants.

2.1.2 Simulation

Un simulateur DEVS d'un modèle couplé se construit sur la structure du modèle en associant un simulateur à chaque modèle atomique et un coordinateur à chaque modèle couplé qui compose le modèle considéré. Un coordinateur particulier, dit coordinateur-racine, gère les relations entre l'environnement et le modèle couplé. Chaque coordinateur oriente les différents messages vers ses coordinateurs-fils et ses simulateurs-fils. Il met à jour ses variables tl et tn (*time last event* et *time next event*) en fonction des messages reçus, et gère une liste d'événements de type (composant d , tn_d) pour envoyer un *-message au coordinateur ou au simulateur concerné par le prochain événement.

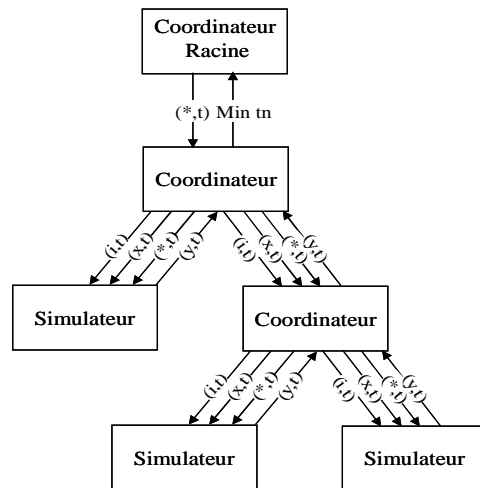


Figure 1. Structure du simulateur DEVS

2.2. Min-Max DEVS

Le formalisme Min-Max DEVS (Giambiasi et Ghosh, 2001) (Hamri et al., 2006) est une extension du formalisme DEVS qui a pour objet de simuler des modèles, dont la durée de vie des états transitoires est spécifiée par un intervalle de temps. En fait, il s'agit de modéliser et simuler des systèmes pour lesquels il est impossible de déterminer avec une certaine précision (ce qui est généralement le cas) la durée de vie des états transitoires. L'intervalle de temps définissant la durée de vie d'un état est alors borné par la durée de vie d'un état transitoire dans le système le plus rapide et celle dans le système le plus lent. Par suite, le modèle devra indiquer que l'état du système réel est inconnu dans cet intervalle de temps.

Le concept de retard min-max a été initialement introduit dans le domaine de la simulation des circuits (Giambiasi, 1980 et VHDL, 1993) pour représenter l'imprécision sur la connaissance des retards réels. D'autres approches ont également été proposées (probabiliste (Magnhagen, 1977), floue (Giambiasi et al., 1994)) mais elles conduisent à des durées de simulation extrêmement plus longues et sont peu utilisables, en pratique, pour des systèmes réels.

Pour plus de clarté, nous rappelons ici brièvement l'interprétation des retards min-max, interprétation qui sera transposée aux Min-Max DEVS.

Considérons un retard min-max associé à une porte logique. La valeur minimale représente l'instant où la porte réelle la plus rapide commutera, la maximale, l'instant où toutes les portes possibles auront commutées. Par suite l'état du modèle entre ces deux instants est inconnu (valeur Φ).

Considérons une porte avec un retard min-max : $r_{\min} = 3$ unités de temps (u.t) et $r_{\max} = 8$ u.t. La figure ci-dessous schématise la réponse à un signal d'entrée ($0 \rightarrow 1$ à $t = 10$ u.t) et ($1 \rightarrow 0$ à $t = 50$ u.t).

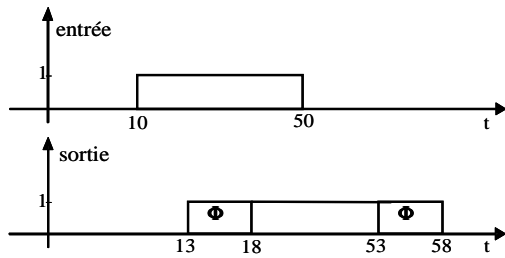


Figure 2. Retard ambigu

A la date 13 u.t, la porte la plus rapide commute, à 18 u.t, toutes les portes ont commuté. Entre 13 et 18, certaines portes sont à la valeur logique 1, les autres sont encore à 0.

En fait, on simule, en une seule passe (réplication), toutes les portes réelles possibles de ce type.

2.2.1 Formalisme Min-Max DEVS

Le concept de retard min-max se transpose au niveau du formalisme DEVS par des durées de vies des états définies par un intervalle de temps à la place de la valeur précise qui est donnée dans un DEVS classique.

Par suite, la spécification externe (utilisateur) associée à chaque état actif une durée de vie minimale et une maximale. Ceci constitue la seule différence, au niveau de la spécification externe, avec un DEVS classique.

Une spécification externe d'un modèle Min-Max DEVS est une structure :

$$ME = (X, Y, S, \delta_{\text{ext}}, \delta_{\text{int}}, \lambda, D)$$

- X, Y, S sont identiques à ceux définis dans le formalisme DEVS (cf. 2.1),
- Les fonctions de transition externe et interne δ_{ext} et δ_{int} , et de sortie λ sont les mêmes que celles définies dans le formalisme DEVS (cf. 2.1), et
- La fonction de durée de vie de l'état $D : S \rightarrow \mathbb{R}^+ \times \mathbb{R}^+$, $D(s_i) = [d_{\min}, d_{\max}]$ avec :
 - d_{\min} est le temps minimal pendant lequel le modèle peut rester dans l'état s_i , il correspond au système réel le plus rapide,
 - d_{\max} est le temps maximal pendant lequel le modèle peut rester dans l'état s_i , il correspond au système réel le plus lent.

Remarque : la spécification externe du modèle Min-Max DEVS est identique à celle du formalisme RT-DEVS (Seong et al., 1998). Cependant, la sémantique opératoire et l'interprétation sont différentes du formalisme

proposé ici. Le formalisme RT-DEVS a été introduit dans le but de réaliser des simulations 'temps réel'.

2.2.2 Sémantique opératoire du formalisme Min-Max DEVS

La simulation d'un modèle Min-Max DEVS repose sur la construction d'un modèle interne (MI). Ce dernier est défini par une structure algébrique semblable à DEVS, mais ses éléments sont augmentés. Nous nous sommes inspirés des modèles à retard ambigu employés dans la modélisation des circuits électroniques pour doter cette structure d'une sémantique propre aux modèles Min-Max DEVS (Giambiasi et Ghosh, 2001) (Hamri et al., 2006).

Ci-dessous nous rappelons brièvement cette structure :

$$MI = (XI, YI, SI, \delta I_{\text{ext}}, \delta I_{\text{int}}, \lambda I, DI) \text{ avec}$$

- $XI = X \times \{\text{fast, slow}\}$ tel que
 - l'événement (x_i, fast) correspond à un événement d'entrée dont la valeur est x_i , il représente (comme nous allons le voir par la suite) l'événement destiné au système le plus rapide, sa date d'occurrence définit le début d'une fenêtre temporelle dans laquelle l'événement x_i peut être reçu (dépendant de la valeur réelle du retard).
 - l'événement (x_i, slow) correspond à un événement d'entrée dont la valeur est x_i . Il représente l'événement destiné au système le plus lent, sa date d'occurrence permet de clore la fenêtre temporelle ouverte par l'événement rapide.

- $SI = (S \cup \{\Phi\}) \times (S \cup \{\Phi\}) \times A$ où :

Φ est la valeur inconnue comme dans les modèles à retard ambigu.

$A = \{\text{external, fast_autonomous, slow_autonomous, passive}\}$

En fonction de la valeur du troisième élément du triplet, nous interprétons un modèle interne dans un état (s_i, s_j, a) tels que $s_i, s_j \in S$ et $a \in A$, comme suit :

- $a = \text{external}$, le modèle a changé d'état suite à l'occurrence d'un événement externe de type rapide (x, fast) . Ce changement concerne seulement le système le plus rapide, les systèmes lents attendent encore l'occurrence de l'événement (x, slow) . Nous déduisons donc que certains systèmes rapides sont déjà dans l'état s_j et d'autres (les plus lents) sont encore dans l'état précédent s_i .
- $a = \text{fast_autonomous}$, les systèmes rapides ont changé d'état par une transition interne car il n'y a pas eu d'occurrence d'événement externe qui aurait annulé la transition interne prévue. De plus, certains systèmes rapides sont dans l'état s_j et d'autres sont encore dans l'état s_i jusqu'au franchissement de la transition interne prévue.
- $a = \text{slow_autonomous}$, tous les systèmes ont changé d'état et la transition interne du système le plus lent a succédé à la dernière transition interne des systèmes rapides. Tous les systèmes ont exécuté la même transition interne pour atteindre l'état futur s_j . Par conséquent, le modèle interne est dans l'état $(s_j, s_j, \text{slow_autonomous})$ si le futur état s_i dans lequel tous les systèmes résident est actif.

- a = passive, l'interprétation d'un état $(s_i, s_i, \text{passive})$ est que cet état est passif et tous les systèmes sont dans l'état s_i .

Partant de ces interprétations, l'ensemble des états possibles du modèle interne est :

$SI = \{(s_i, s_i, \text{passive}), (s_i, s_i, \text{slow_autonomous}), (s_i, s_j, \text{fast_autonomous}), (s_i, s_j, \text{external}), (\Phi, \Phi, \text{passive}) \mid s_i, s_j \in S\}$, les autres états sont exclus.

L'état totalement inconnu $(\Phi, \Phi, \text{passive})$ représente le fait qu'à l'instant considéré il y a plus de deux états possibles pour le système réel.

• $YI = YX\{\text{fast, slow}\}$ où :

- $((y_i, \text{fast}), t_i)$ est un événement de sortie dont la date d'occurrence est t_i , il représente la réponse du système réel le plus rapide.

- $((y_j, \text{slow}), t_j)$ est un événement de sortie dont la date d'occurrence est t_j , il représente la réponse du système réel le plus lent.

Ainsi nous définissons les fonctions suivantes :

$$\delta I_{\text{ext}} : QI \times XI \rightarrow SI,$$

$$\delta I_{\text{int}} : SI \rightarrow SI,$$

$$\lambda I : SI \rightarrow YI,$$

$$DI : SI \rightarrow R^+.$$

La construction détaillée de ce modèle interne est donnée dans (Giambiasi et Ghosh, 2001) et (Hamri et al., 2006).

3 MIN-MAX DEVS COUPLES

Afin de permettre une description hiérarchique des modèles Min-Max DEVS, la fermeture sous couplage du formalisme doit être démontrée.

Un modèle Min-Max DEVS couplé est défini par la même structure qu'un DEVS couplé classique.

3.1 Fermeture sous couplage du Min-Max DEVS

Cette démonstration consiste à prouver que pour tout modèle Min-Max DEVS couplé il existe un modèle Min-Max DEVS atomique équivalent. Grâce à cette démonstration un modèle hiérarchique Min-Max DEVS peut être mis à plat et couplé avec d'autres modèles atomiques.

Dans ce qui suit, la démonstration repose sur la définition abstraite des modèles Min-Max DEVS couplés par un réseau de modèles atomiques dont le concept de port n'est pas utilisé pour décrire l'interface. Nous rappelons ci-dessous la définition d'un tel réseau, qui est la même pour un réseau Min-Max DEVS. Un réseau DEVS N est défini par la structure suivante :

$$N = (X, Y, D, \{M_d\}, \{I_d\}, \{Z_{i,d}\}, \text{select}), \text{ où}$$

X : est l'ensemble des événements externes d'entrée,

Y : est l'ensemble des événements externes de sortie,

D : est l'ensembles des composants,

M_d : est un modèle DEVS classique pour chaque $d \in D$,

I_d : est l'ensemble des influenceurs de d tel que $I_d \subseteq$

$$D \cup \{N\} \text{ pour chaque } d \in D, \text{ et}$$

$Z_{i,d}$: est la fonction de transformation de i-vers-d avec

$$Z_{i,d} : X \rightarrow X_d \text{ si } i = N$$

$$Z_{i,d} : Y_d \rightarrow Y \text{ si } d = N$$

$$Z_{i,d} : Y_i \rightarrow X_d \text{ si } d \neq N \text{ et } i \neq N.$$

Considérons maintenant un réseau de modèles Min-Max DEVS atomiques désigné par N.

Le modèle Min-Max DEVS atomique équivalent au réseau N est défini par la structure ci-dessous dont nous allons définir chaque élément.

$$\text{Min-Max DEVS}_N = (X, Y, S, \delta_{\text{ext}}, \delta_{\text{int}}, \lambda, D).$$

L'interface du modèle atomique correspond à l'interface du réseau N. En effet les événements reçus par le modèle N doivent être reçus par le modèle atomique équivalent. Il est de même pour les événements de sortie.

L'ensemble des états défini dans le modèle atomique équivalent est défini par l'ensemble des états de chaque modèle atomique qui compose le réseau N ($d \in D$). Cet ensemble est défini comme suit :

$$S = \times_{d \in D} Q_d \text{ avec } Q_d = \{(s, e) \mid s \in S_d \text{ et } e \in R^+\} \text{ et}$$

$$D : S \rightarrow R^+ \times R^+ \text{ avec } D(s) = [\min(\min(D_0(s_0)), \dots, \min(D_d(s_d)), \dots), \min(\max(D_0(s_0)), \dots, \max(D_d(s_d)), \dots)]$$

Cette dernière expression indique qu'une transition interne peut être franchie dans un modèle atomique d au plus tôt au minimum des minimums et au plus tard au minimum des maximums des durées de vie des états décrits. Ainsi une transition est franchie dans une fenêtre temporelle définie par une valeur min-max. Si la durée minimale est égale à zéro, la transition peut être franchie immédiatement.

Par suite, nous pouvons définir le temps restant avant le franchissement de la prochaine transition dans le modèle atomique équivalent à N, en tenant compte du temps écoulé e dans l'état en cours.

$$\sigma = [\min(D(s)) - e, \max(D(s)) - e] \text{ si } \min(D(s)) - e \geq 0$$

$$\sigma = [0, \max(D(s)) - e] \text{ sinon.}$$

Donc, La prochaine transition peut être franchie dans une fenêtre $[\sigma_{\min}, \sigma_{\max}]$.

Dans certains cas, il peut y avoir plusieurs transitions internes à franchir dans des modèles atomiques différents à une date donnée. Afin d'identifier laquelle devra être activée en premier, nous définissons l'ensemble des modèles candidats à la prochaine transition interne comme suit :

$$\text{IMM}(s) = \{d \in D \mid \text{Min}(\sigma_d) = \text{Min}(D(s))\}.$$

Ne pouvant activer qu'une seule transition à la fois, le modèle concerné est choisi grâce à la fonction 'select', comme suit :

$$d^* = \text{select}(\text{IMM}(s)).$$

Par suite, la fonction de transition interne du modèle atomique équivalent est définie par :

$$\delta_{\text{int}} : S \rightarrow S$$

$$\delta_{\text{int}}(s) = s' = (\dots, (s_d', e_d'), \dots) \text{ avec}$$

$$(s_d', e_d') = (\delta_{\text{int}, d}(s_d), 0) \text{ si } d = d^*$$

$$(s_d', e_d') = (\delta_{\text{ext}, d}(s_d, e_d + e_{d^*}, Z_{d^*, d}(\lambda(s_{d^*}))), 0) \text{ si } d^* \in I_d \text{ et } \lambda(s_{d^*}) \neq \phi$$

$$(s_d', e_d') = (s_d, e_d + e_{d^*}) \text{ sinon.}$$

Lors de l'activation d'une transition interne, un événement de sortie est émis par le modèle atomique d^* . La fonction de sortie du modèle atomique équivalent doit donner le même événement que le modèle atomique d^* . La fonction de sortie du modèle équivalent est donc définie comme suit :

$$\lambda(s) = Z_{d^*, N}(\lambda(s_{d^*})) \text{ si } d^* \in I_N$$

$$\lambda(s) = \phi \text{ sinon.}$$

Lorsque le réseau N reçoit un événement externe, il le transmet au modèle atomique concerné en utilisant la fonction d'interface ($Z_{N,d}$). Ce dernier déclenche une transition externe, suite à cet événement, pour définir son nouvel état. La fonction de transition externe du modèle atomique équivalent au réseau N doit effectuer le même changement d'état. Les autres modèles atomiques n'ayant pas reçu cet événement demeurent dans le même état. Par suite, la fonction de transition externe du modèle atomique équivalent δ_{ext} est définie comme suit :

$$\delta_{\text{ext}}: Q \times X \rightarrow S$$

$$\delta_{\text{ext}}((s, e), x) = s' = (\dots, (s_d', e_d'), \dots) \text{ où}$$

$$(s_d', e_d') = (\delta_{\text{ext}}((s_d, e_d + e), x), 0) \text{ si } N \in I_d$$

$$(s_d', e_d') = (s_d, e_d + e) \text{ sinon}$$

L'ensemble de ces définitions nous permet de conclure que le modèle équivalent à un modèle Min-Max DEVS couplé est un modèle Min-Max DEVS atomique.

Exemple : Considérons un modèle DEVS couplé N composé des deux modèles atomiques A et B :

$$A = (X_A, Y_A, S_A, \delta_{\text{int}, A}, \delta_{\text{ext}, A}, \lambda_A, D_A)$$

$$X_A = \{x \mid x \in \mathbb{R}^+\}$$

$$Y_A = \{y \mid y \in \mathbb{R}^+\}$$

$$S_A = \{s_0, s_1\}$$

$$\delta_{\text{int}, A}(s_0) = s_1$$

$$\delta_{\text{ext}, A}(s_1, e, x) = s_0$$

$$\delta_{\text{ext}, A}(s_0, e, x) = s_0$$

$$D_A(s_0) = [5, 10]$$

$$D_A(s_1) = \text{infini}$$

$$\lambda_A(s_0) = y$$

$$B = (X_B, Y_B, S_B, \delta_{\text{int}, B}, \delta_{\text{ext}, B}, \lambda_B, D_B)$$

$$X_B = \{x \mid x \in \mathbb{R}^+\}$$

$$Y_B = \{y \mid y \in \mathbb{R}^+\}$$

$$S_B = \{s_0, s_1\}$$

$$\delta_{\text{int}, B}(s_0) = s_1$$

$$\delta_{\text{ext}, B}(s_1, e, x) = s_0$$

$$\delta_{\text{ext}, B}(s_0, e, x) = s_0$$

$$D_B(s_0) = [8, 9]$$

$$D_B(s_1) = \text{infini}$$

$$\lambda_B(s_0) = y$$

$$N = (X_N, Y_N, D, \{M_{d \in D}\}, \{I_d\}, Z_{i,d}, \text{select})$$

$$X_N = X_A \text{ et } Y_N = Y_B$$

$$D = \{A, B\}$$

$$I_A = \{N, B\}, I_B = \{A\} \text{ et } I_N = \{B\}$$

$$Z_{N,A} : X_N \rightarrow X_A, Z_{B,N} : Y_B \rightarrow X_N, Z_{B,A} : Y_B \rightarrow X_A \text{ et}$$

$$Z_{B,N} : Y_B \rightarrow Y_N$$

Nous construisons le modèle équivalent N_{eq} au réseau de modèles N, comme suit :

$$N_{\text{eq}} = (X_{\text{eq}}, Y_{\text{eq}}, S_{\text{eq}}, \delta_{\text{int}, \text{eq}}, \delta_{\text{ext}, \text{eq}}, \lambda_{\text{eq}}, D_{\text{eq}})$$

$$X_{\text{eq}} = X_N = X_A \text{ et } Y_{\text{eq}} = Y_N = Y_B$$

$$S_{\text{eq}} = \{((s_A, e_A), (s_B, e_B)) \mid s_A \in S_A, s_B \in S_B, e_A, e_B \in \mathbb{R}^+\}$$

$$= \{((s_0, e_A), (s_0, e_B)), ((s_1, e_A), (s_1, e_B)), ((s_0, e_A), (s_1, e_B)), ((s_1, e_A), (s_0, e_B)) \mid e_A, e_B \in \mathbb{R}^+\}$$

$$\delta_{\text{int}, \text{eq}}((s_0, e_A), (s_0, e_B)) = ((s_1, 0), (\delta_{\text{ext}, B}(s_0, e_B + e_A, x), 0))$$

$$= ((s_1, 0), (s_1, 0))$$

Le modèle A produit une sortie y qui va influencer immédiatement le modèle B. D'où l'activation de la

fonction $\delta_{\text{ext}, B}$, pour le calcul du nouvel état de N_{eq} . Par conséquent le temps écoulé dans chaque état est remis à zéro après cette transition ($\delta_{\text{int}, \text{eq}}$). De même :

$$\delta_{\text{int}, \text{eq}}((s_1, e_A), (s_1, e_B)) = ((\delta_{\text{ext}, B}(s_0, e_A + e_B, x), 0), (s_0, 0))$$

$$= ((s_0, 0), (s_0, 0)).$$

$\delta_{\text{ext}, \text{eq}}(((s_1, e_A), (s_1, e_B)), e, x) = ((\delta_{\text{ext}, A}(s_0, e, x), 0), (s_0, e_B + e))$. Nous appliquons cette transition externe au modèle A puisqu'il est influencé par N ($N \in I_A$).

Dans cet exemple seul le modèle B envoie un événement de sortie au modèle N, d'où :

$$\lambda_{\text{eq}}((s_1, e_A), (s_1, e_B)) = Z_{B,N}(\lambda_B((s_1)))$$

$$D((s_0, e_A), (s_0, e_B)) = [5, 10] \text{ et}$$

$$D((s_1, e_A), (s_1, e_B)) = [8, 9]. \text{ Les deux autres états sont inatteignables.}$$

4 SIMULATION

La sémantique opératoire des Min-Max DEVS étant différentes des DEVS classiques, nous définissons un simulateur conceptuel de modèles Min-Max DEVS atomiques, puis le coordinateur qui permet la simulation de modèles couplés.

Le simulateur s'appuie sur la représentation interne qui a été déduite de la spécification de l'utilisateur. La simulation, comme pour les DEVS classiques, s'effectue par échange de messages, qui sont de 3 types :

- i-message (i, t) pour initialiser le modèle,
- *-message (*, t) pour une transition interne conduisant à l'exécution des fonctions de sortie λI et de transition interne δI_{int} , et
- deux messages (x, fast)-message et (x, slow)-message pour activer la fonction de transition externe. Bien que les deux messages conduisent à l'exécution de la même fonction, nous avons personnalisé ces messages en fonction de leur contenu.

L'état du modèle est représenté par un triplet (s, s, a) tel que $s \in S$ et $a \in A$. Cet état (s, s, a) représente l'état courant du modèle interne et par conséquent les états possibles du modèle externe Min-Max DEVS.

Le simulateur manipule, en plus des variables du modèle, 3 variables :

- tl qui sauvegarde la date d'occurrence du dernier événement reçu, il peut s'agir d'un événement externe rapide ou lent (x, fast or slow) ou d'un événement interne.
- La variable tn représentant la date de la prochaine transition interne, date qui sera transmise au coordinateur parent. Elle dépend de la variable tl et de l'état courant du modèle interne :

$$tn = tl + DI(s) \mid s \in SI.$$
- La variable e définit le temps écoulé dans l'état courant depuis la dernière transition du modèle interne. Connaissant la date courante t, e s'obtient simplement par :

$$e = t - tl.$$

Grâce au modèle interne une simulation déterministe peut être effectué. Le simulateur active les fonctions du modèle interne suivant le type de messages reçus. En Min-Max DEVS, il existe deux types de x-messages correspondant aux événements externes rapides et aux

lents. Il y a un seul type de *-messages bien qu'il existe deux transitions internes différentes. La sémantique opératoire définie par le modèle interne permet de distinguer les deux transitions en tenant compte de l'état en cours.

```

Min-Max-DEVS_Simulator
Variables
  tl, tn, e: Real
  s: State variable | s ∈ SI
  parent: coordinator
  EM: External Model
  IM: Internal Model
//s0 ∈ S the initial state of EM
when receive i-message (i, t0) at time t0
  if (s0 is passive) then s = (s0, s0, pas-
  sive)
    else s = (s0, s0, slow_autonomous)
  tl = t0
  tn = tn + DI(s)
when receive *-message (*, t) at time t
  y = λI(s)
  send y-message (y, t) at time t
  s = δIint(s)
  tl = t
  tn = t + DI(s)
when receive x-message ((x, fast), t) or ((x,
  slow), t) at time t
  e = t - tl
  if (x is a fast event) then s = δIext(s, e,
  (x, fast))
    else s = δIext(s, e, (x, slow))
  tl = t
  tn = t + DI(s)
end Min-Max-DEVS-simulator
    
```

Algorithme 1. Simulateur Min-Max DEVS

Au début de la simulation, le simulateur Min-Max DEVS reçoit un i-message pour activer l'état initial $S_0 \in SI$. Tenant compte de la nature de cet état, il déduit la date de la prochaine transition interne, si elle existe. L'état S_0 peut être passif ou actif de forme $(s_0, s_0, \text{slow_autonomous})$ s'il est actif. Dans ce dernier cas, la durée de vie de cet état est $DI(s_0, s_0, \text{slow_autonomous}) = \text{Min}(D(s_0))$.

Par suite, le simulateur affecte respectivement aux variables tl and tn les valeurs t_0 et $t_0 + DI(S_0)$. Le modèle externe est à l'état s_0 au début de la simulation et il devient indéfini entre les deux dates $[t_0 + \text{Min}(D(s_0)), t_0 + \text{Max}(D(s_0))]$, si aucun événement externe ne survient. Le coordinateur parent informe le simulateur sur l'occurrence de la prochaine transition externe ou interne par un x-message ou un *-message, en fonction de sa liste d'événements.

Lorsque le simulateur reçoit un *-message $(*, t)$, il active les fonctions de sortie λI et de transition interne δI_{int} pour émettre l'événement de sortie et déduire le futur état. Ensuite, il met à jour ses variables tl et recalcule $tn = tl + DI(\text{nouvel état})$.

Dans le cas d'un message externe (x-message (x, t)), le simulateur distingue l'événement rapide du lent pour activer correctement la fonction de transition externe δI_{ext} . Les variables tl et tn sont mises à jour en leur affectant les nouvelles valeurs t et $t + DI(\text{nouvel état})$.

Le coordinateur Min-Max DEVS diffère du coordinateur DEVS seulement par le format des messages reçus et

envoyés qui sont adaptés aux événements d'entrée et de sortie du modèle interne MI. Il envoie les événements externes au simulateur Min-Max DEVS pour activer la fonction de transition externe δI_{ext} , et il transforme les événements de sortie (fonction λI) en événements d'entrée pour les successeurs du modèle traité.

5 EXEMPLE

5.1 Description du système

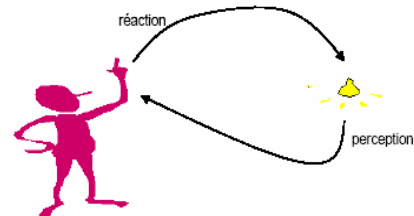


Figure 3. Illustration de l'exemple

Considérons une lampe (Lumière) commandée par une personne (ActeurLumière). La personne allume la lumière dès qu'elle s'aperçoit qu'elle est éteinte. Le délai de sa réaction varie d'un essai à l'autre. L'analyse de ces délais permet de définir l'intervalle de valeurs (1.5 u.t, 2 u.t). La durée d'éclairage dépend de différents paramètres : procédé de fabrication, matières utilisées, etc. Pour plus de simplicité nous considérons, dans cet exemple, une valeur unique (5 u.t).

Un modèle DEVS classique conduit à représenter le délai de réaction par une moyenne des valeurs obtenues (1.75 u.t). On obtient alors les deux modèles DEVS classiques suivants :

$$\text{ActeurLumière} = \text{Act}_{\text{DEVS}} = (X, Y, S, \delta_{\text{int}}, \delta_{\text{ext}}, \lambda, D)$$

$$\begin{aligned}
 X &= \{(In, OFF)\} \\
 Y &= \{(Out, ON)\} \\
 S &= \{\text{Aperçu, réagi}\} \\
 \delta_{\text{int}}(\text{aperçu}) &= \text{réagi} \\
 \delta_{\text{ext}}(\text{réagi}, e, In ? OFF) &= \text{aperçu} \\
 \lambda(\text{aperçu}) &= (Out ! ON) \\
 D(\text{aperçu}) &= 1.75 \\
 D(\text{réagi}) &= \text{infini}
 \end{aligned}$$

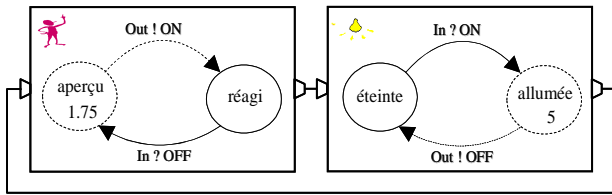
$$\text{Lumière} = \text{Lum}_{\text{DEVS}} = (X, Y, S, \delta_{\text{int}}, \delta_{\text{ext}}, \lambda, D)$$

$$\begin{aligned}
 X &= \{(In, ON)\} \\
 Y &= \{(Out, OFF)\} \\
 S &= \{\text{éteinte, allumée}\} \\
 \delta_{\text{int}}(\text{allumée}) &= \text{éteinte} \\
 \delta_{\text{ext}}(\text{éteinte}, e, In ? ON) &= \text{allumée} \\
 \lambda(\text{allumée}) &= (Out ! OFF) \\
 D(\text{éteinte}) &= \text{infini} \\
 D(\text{allumée}) &= 5
 \end{aligned}$$

$$\text{LumièreCouplée} = (X, Y, D, \{M_d \mid d \in D\}, \text{EIC}, \text{EOC}, \text{IC})$$

$$\begin{aligned}
 X &= Y = \{\} \\
 D &= \{\text{Act}_{\text{DEVS}}, \text{Lum}_{\text{DEVS}}\} \\
 \text{EIC} &= \text{EOC} = \{\} \\
 \text{IC} &= \{((\text{Act}_{\text{DEVS}}, \text{Out}), (\text{Lum}_{\text{DEVS}}, \text{In})), ((\text{Lum}_{\text{DEVS}}, \text{Out}), (\text{Act}_{\text{DEVS}}, \text{In}))\}
 \end{aligned}$$

a) Modèles DEVS classiques



b) Description graphique des modèles DEVS

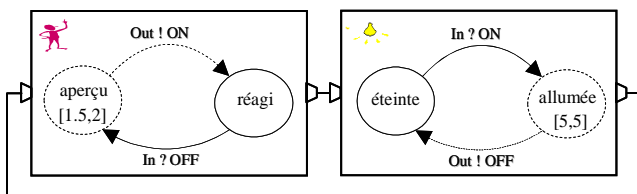
Figure 4. Modélisation de l'exemple avec DEVS

L'approximation du délai de réaction par une valeur moyenne conduit à des résultats de simulation qui représentent mal la réalité. Par exemple, la réaction pouvant avoir lieu avant $t = 1.75$ u.t, la lumière peut être rallumée avant ce délai, ce qui n'est pas indiqué par cette modélisation à valeurs précises.

Le modèle Min-Max DEVS permet de représenter l'imprécision du délai de réaction de la personne par une durée de vie de l'état $D(\text{réaction}) = [1.5, 2]$. Nous obtiendrons alors les modèles Min-Max DEVS suivants :

ActeurLumière (Modèle Externe)=
 $Act_{ME} = (X, Y, S, \delta_{int}, \delta_{ext}, \lambda, D)$
 $X = \{(In, OFF)\}$
 $Y = \{(Out, ON)\}$
 $S = \{\text{aperçu, réagi}\}$
 $\delta_{int}(\text{aperçu}) = \text{réagi}$
 $\delta_{ext}(\text{réagi}, e, In ? OFF) = \text{aperçu}$
 $\lambda(\text{aperçu}) = (Out !ON)$
 $D(\text{aperçu}) = [1.5, 2]$
 $D(\text{réagi}) = \text{infini}$
 Lumière (Modèle Externe) = $Lum_{ME} = (X, Y, S, \delta_{int}, \delta_{ext}, \lambda, D)$
 $X = \{(In, ON)\}$
 $Y = \{(Out, OFF)\}$
 $S = \{\text{éteinte, allumée}\}$
 $\delta_{int}(\text{allumée}) = \text{éteinte}$
 $\delta_{ext}(\text{éteint}, e, In ? ON) = \text{allumée}$
 $\lambda(\text{allumée}) = (Out !OFF)$
 $D(\text{éteinte}) = \text{infini}$
 $D(\text{allumée}) = [5, 5]$
 LumièreCoupée = $(X, Y, D, \{M_d | d \in D\}, EIC, EOC, IC)$
 $X = Y = \{\}$
 $D = \{Act_{ME}, Lum_{ME}\}$
 $EIC = EOC = \{\}$
 $IC = \{((Act_{ME}, Out), (Lum_{ME}, In)), ((Lum_{ME}, Out), (Act_{ME}, In))\}$

a) Modèles Min-Max DEVS



b) Description graphique des modèles Min-Max DEVS

Figure 5. Modélisation de l'exemple avec Min-Max DEVS

5.2 Simulation

Initialement, les modèles internes Act_{MI} et Lum_{MI} sont respectivement dans l'état (aperçu, aperçu,

slow_autonomous) et (éteinte, éteinte, slow_autonomous). Nous avons opté pour initialiser les réactions rapide et tardive de la personne à la même date $t = 0$ (dans le but de simplifier la simulation de cet exemple). Une fois les modèles initialisés, le simulateur Act informe le coordinateur de sa prochaine transition ($t = 1.5$ u.t). Ce dernier envoie un *-message à la date correspondante pour activer les fonctions de sortie et de transition interne. Ensuite, le simulateur Act envoie un y-message correspondant à l'événement de sortie ((Out ! ON), fast). Le modèle interne Act_{MI} transite vers l'état (aperçu, réagi, external) à la date $t = 1.5$ u.t. Cet état étant actif, une transition interne est prévue à la date $t_n = t_l + DI(S_{Act}) = 1.5 + 0.5 = 2$ u.t. Le coordinateur transforme le y-message reçu en x-message pour l'envoyer au simulateur Lum. Celui-ci active la fonction de transition externe pour mettre à jour l'état du modèle interne Lum_{MI} qui devient (éteinte, allumée, external). Cet état étant actif, une transition interne est prévue à la date $t_n = t_l + D(S_{Lum}) = 1.5 + 5 = 6.5$ u.t.

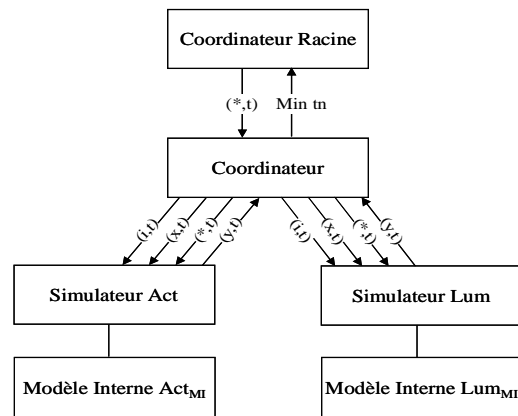


Figure 6. Architecture du simulateur Min-Max DEVS

Le coordinateur retire de sa liste d'événements le prochain événement à traiter, qui est un *-message de date $t_n = 2$ u.t, pour le simulateur Act. Ce dernier exécute les fonctions de sortie et de transition interne du modèle interne Act_{MI} . En conséquence, il émet un y-message avec un événement de sortie ((Out ! ON), slow) de date $t = 2$ u.t. Le modèle transite vers l'état (réagi, réagi, passive). Cet état est passif et aucune transition interne future n'est prévue.

Au cinquième cycle le modèle interne Act_{MI} atteint l'état totalement inconnu (Φ, Φ , passive), et il n'influence plus le modèle interne Lum_{MI} . Ce dernier réside définitivement dans l'état passif (éteinte, éteinte, passive).

5.3 Analyse des résultats

Au début, la lumière est éteinte et la personne ne réagit pas immédiatement. La durée de sa réaction est comprise entre 1.5 et 2 u.t. Chaque valeur appartenant à cet intervalle, représenté par la durée de vie de l'état (aperçu, réagi, external), est une valeur possible de la date de réaction. Cet état modélise l'ensemble des réactions possibles qui sont comprises entre une réaction rapide à la

date $t = 1.5$ u.t et une tardive à la date $t = 2$ u.t. Par suite, la lumière s'éteint à une date comprise dans l'intervalle $[6.5, 7$ u.t.]. Cette durée imprécise est due à la réaction de la personne.

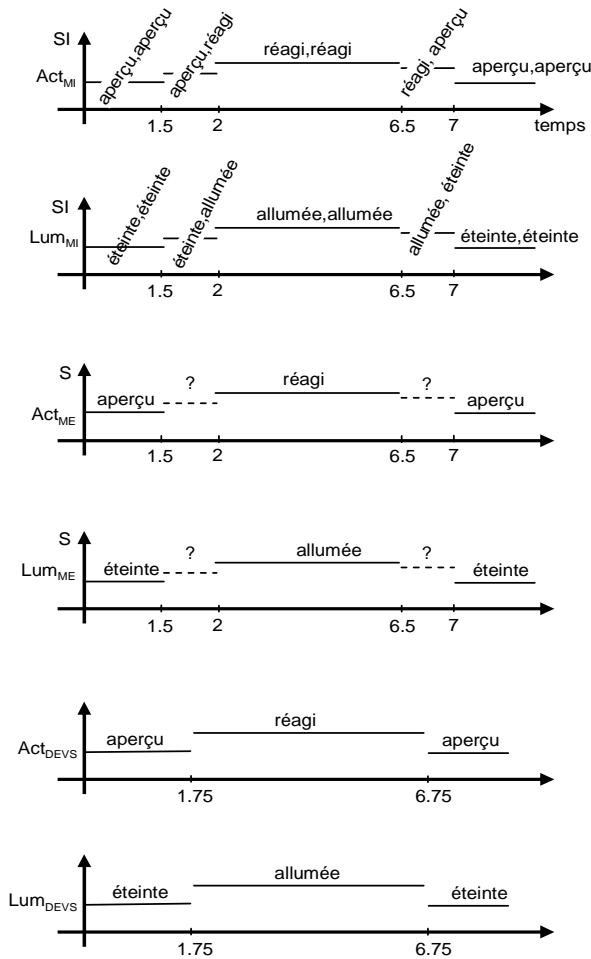


Figure 7. Trace de simulation des différents modèles Min-Max DEVS et DEVS pour le 1^{er} cycle réaction-perception

Après la cinquième réaction, le modèle atteint l'état totalement inconnu (Φ, Φ , passive). Cet état indique qu'il existe plus de deux états possibles pour le système réel à l'instant considéré. La conclusion est alors que l'imprécision sur les temps de réactions ne permet pas de connaître l'état du système réel. Cette conclusion est à comparer avec celle que l'on obtiendrait sur un modèle à valeurs précises qui lui indiquerait un état précis de la lumière, état qui, dans de nombreuses situations, ne correspondrait pas à la réalité.

6 CONCLUSION

Le formalisme Min-Max DEVS permet la construction et la simulation de modèles plus proches de la réalité en permettant de prendre en compte l'imprécision sur les durées de vie des états transitoires.

De plus, la fermeture sous couplage simplifie la construction de modèles complexes par réutilisation de modèles de base, préalablement stockés en bibliothèque.

La simulation de modèles Min-Max DEVS repose sur la même structure que celle des DEVS classiques (simulateur, coordinateur et coordinateur racine). La seule différence concerne le format des x-messages que nous avons modifié pour assurer un traitement adapté aux différents types d'événements externes.

Le simulateur Min-Max DEVS peut être remplacé par un simulateur G-DEVS (Giambiasi et al., 2001). En effet, en G-DEVS, un événement externe est défini par un vecteur de valeurs ce qui permet de représenter directement un événement externe en Min-Max DEVS.

Nos travaux futurs consisteront à intégrer ce nouveau formalisme dans le standard DEVS, en définissant un méta-formalisme qui regroupe les différentes extensions que nous proposons.

REFERENCES

- Giambiasi N. « Contribution à la conception assistée des systèmes digitaux » Thèse de doctorat d'état, Université des Sciences et Techniques du Languedoc, Montpellier, 1980.
- Giambiasi N., M. Smaili et C. Frydman « Discrete event simulation with fuzzy dates » Conférence Proceedings *European Simulation Symposium 1994*, Istanbul, Turkey.
- Giambiasi N., N. Escudé et S. Ghosh « Generalized Discrete Event Simulation of Dynamic Systems », in: *Issue 4 of SCS Transactions: Recent Advances in DEVS Methodology-part II*, vol. 18, pp. 216-229, décembre 2001. ISSN:0740-6797.
- Giambiasi, N. et S. Ghosh « Min-Max DEVS: A new formalism for the specification of discrete event models with Min-Max delays ». Conférence Proceedings *European Simulation Symposium*, Marseille, France, 18-20 octobre 2001, pp. 616-621.
- Introduction to VHDL, Mentor Graphics Corporation, 1993.
- Hamri, M., N. Giambiasi et C. Frydman « Min-Max DEVS modeling and simulation ». *Journal simulation modelling practice and theory*. Vol. 14, issue 7 pp. 909-929 octobre 2006.
- Magnhagen B. « Probability based verification of time margins in digital design » Thèse de doctorat d'état, Université Linköping, Suède, 1977.
- Seong, M et T. G. Kim « Real-time DEVS simulation : Concurrent, Time-selective execution of combined RT-DEVS model and interactive environment » *Proceedings of SCSC-98*, 19-22 juillet 1998 Reno, NV, pp 410 -415.
- Zeigler, B. « Theory of modelling and simulation ». Première édition, *Robert E. Kreiger publishing company*, Florida, USA, 1976.
- Zeigler, B. « Multifaceted modeling and discrete event simulation ». Edition Academic Press, Londres, Angleterre, 1984.
- Zeigler, B., H. Praehofer et T. G. Kim « Theory of modeling and simulation ». Deuxième édition *Academic Press*, 2000.