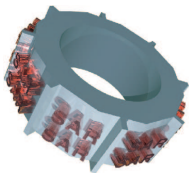
Master UPMC Sciences et technologies, mention informatique
Spécialité Systèmes et Applications Réparties

Modélisation Formelle de Systèmes Répartis



Projet: modélisation et vérification du comportement de véhicules automatiques sur un pont à voie unique

Fabrice.Kordon@lip6.fr

5 novembre 2009

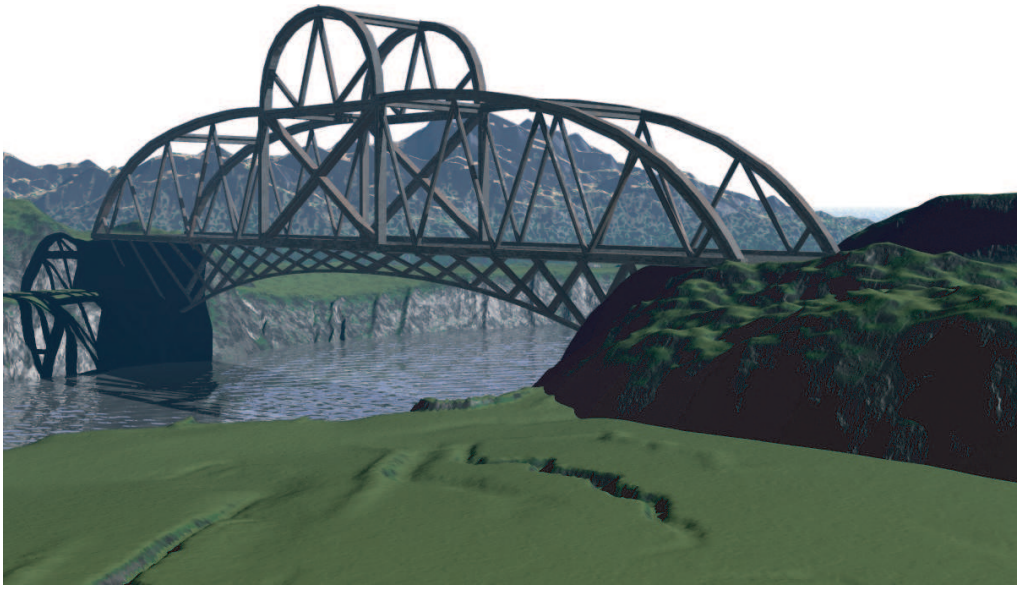


Table des matières

1 Passage de véhicules automatiques sur un pont à voie unique **3**

2 Extension du système pour gérer des véhicules prioritaires **5**

3 Structure de la livraison **6**

Avant-propos

Nous vous conseillons de lire ***attentivement le sujet dans son intégralité*** avant de répondre aux questions. L'examen de la structure du rapport que vous devrez rendre est également à étudier. Le respect de cette structure pour votre rapport vous est demandé (un template LaTeX vous est fourni pour ceux qui n'y sont pas allergique ;-).

Dans ce qui suit, nous vous demandons d'identifier et de justifier toutes les abstractions que vous identifierez dans le système.

Le rapport doit scrupuleusement suivre la structure que nous vous proposons.

1 Passage de véhicules automatiques sur un pont à voie unique

Le système que nous considérons est composé des entités suivantes :
- VAA, qui représente un véhicule automatisé de type *A* (qui circule dans un sens donné sur le pont),
- VAB, qui représente un véhicule automatisé de type *B* (qui circule dans un sens donné sur le pont),
- CTRLP, qui représente le système de contrôle qui autorisera les VAA et les VAB à entrer sur le pont,
- P, qui représente les propriétés intrinsèques du pont.

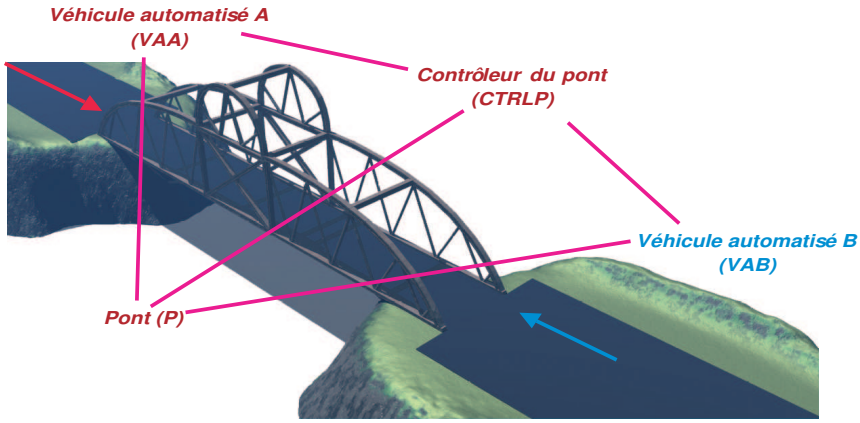


FIG. 1 – Relations entre les entités du système.

Ces entités son reliées suivant le schéma de la Figure 1. Chaque véhicule doit, avant d'entrer sur le pont, communiquer[1] avec le contrôleur associé pour en obtenir l'autorisation. Lorsqu'il obtient cette autorisation, il redémarre pour partir sur le pont ; dans le cas contraire, il se met en attente. Le contrôleur doit quant-à-lui assurer que l'ouverture du pont à une catégorie de véhicule s'effectue de manière équitable, c'est-à-dire soit lorsque N_{alt} véhicules d'un même type sont passés sur le pont, soit à l'expiration d'une temporisation. On considèrera que le système contient N_{VAA} VAA et N_{VAB} VAB. Enfin, la capacité du pont est de $Capa_p$ véhicules, quelque soit leur type.

On propose pour le système l'architecture indiquée en Figure 2. Cette figure spécifie des «services» que l'on attend du système mais sans en détailler la sémantique et les paramètres.

Question 1.1 : interfaces et composants du système

Déterminez, pour chaque composants du système, les interfaces détaillée en indiquant, pour chaque composant, le type d'interface (place ou transition) que vous suggérez. Vous devrez valider votre choix par rapport au cahier des charges qui vous est donné.

[1] On suppose l'existence d'un réseau hertzien fiable aux alentours du pont.

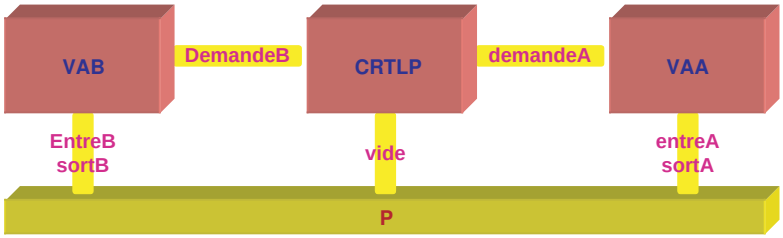


FIG. 2 – Architecture proposée pour le système.

Question 1.2 : modélisation du composant VAA

Modélisez le comportement d'un VAA. Vérifiez que cette modélisation est correcte «localement».

Question 1.3 : modélisation du composant VAB

Modélisez le comportement d'un VAB en vous inspirant du travail réalisé pour le VAA ; vous vérifierez que cette modélisation est également cohérente.

Question 1.4 : modélisation du composant P

Modélisez le comportement des caractéristiques du pont assurées par le composant P. Vérifiez que cette modélisation est correcte «localement».

Question 1.5 : modélisation du composant CTRLP

Modélisez le comportement des caractéristiques du CTRLP. Vérifiez que cette modélisation est correcte «localement».

Question 1.6 : assemblage

Procédez à l'assemblage des composants que vous avez modélisé. Vérifiez que cet assemblage est cohérent et que les «propriétés triviales attendues» sont bien présentes dans cette spécification.

Question 1.7 : vérification

Procédez à la vérification formelle du modèle pour les propriétés suivantes :

P_1 : Il n'y a pas de collision (i.e. deux véhicules circulants en sens inverse) sur le pont.

P_2 : Un véhicule qui arrive est certain de passer sur le pont à l'issue d'une durée bornée.

Vous exprimerez ces propriétés en des termes vérifiables et utiliserez les outils offerts par CPN-AMI pour démontrer leur véracité. Vous vous intéresserez à des valeurs remarquables des paramètres identifiés pour le système : N_{VAA}, N_{VAB}, $Capa_p$ et N_{alt} (en justifiant le choix des rapports entre ces valeurs).

Question 1.8 : retour sur les spécifications du système

Vous proposerez une solution cohérente en termes d'algorithmique répartie pour chacun des composants identifiés dans le système à des fin d'implémentation par une équipe de développeurs. Identifiez les propriétés à préserver et surtout la façon dont vous suggérez qu'elles devront être validées sur le système réel.

2 Extension du système pour gérer des véhicules prioritaires

On considère désormais que le système possède deux nouveaux types d'entités :
– VAPA, qui représente un véhicule automatisé *prioritaire* de type A ; il a les même caractéristiques qu'un VAA mais il doit entrer prioritairement sur le pont (c'est-à-dire avant qu'un véhicule non prioritaire VAA ou VAB n'entre sur le pont).
– VAPB, qui représente un véhicule automatisé *prioritaire* de type B ; il a les même caractéristiques qu'un VAB mais il doit entrer prioritairement sur le pont (c'est-à-dire avant qu'un véhicule non prioritaire VAB ou VAA n'entre sur le pont).

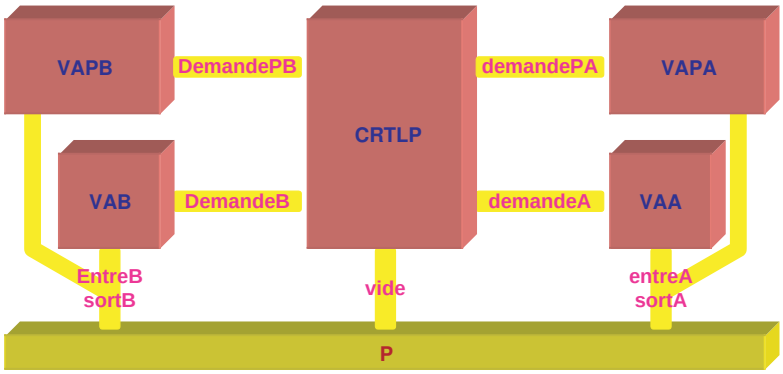


FIG. 3 – Architecture proposée pour le système.

On propose pour cette extension du système l'architecture indiquée en Figure 3. Comme précédemment, cette figure spécifie des «services» que l'on attend du système mais sans en détailler la sémantique et les paramètres.

Question 2.1 : interfaces et composants du système

Modifiez, pour les composants pré-existants dans le système précédent et déterminez, pour les nouveaux composants, les interfaces détaillée en indiquant, le type d'interface (place ou transition) que vous suggérez. Vous devrez valider votre choix par rapport au cahier des charges qui vous est donné.

Question 2.2 : modélisation du composant VAA

Mettez à jour si besoin est le comportement d'un VAA. Vérifiez que cette modélisation est correcte «localement».

Question 2.3 : modélisation du composant VAB

Mettez à jour si besoin est le comportement d'un VAB. Vérifiez que cette modélisation est correcte «localement».

Question 2.4 : modélisation du composant VAPA

Modélisez le comportement d'un VAPA. Vérifiez que cette modélisation est correcte «localement».

Question 2.5 : modélisation du composant VAPB

Modélisez le comportement d'un VAPB en vous aidant de la modélisation d'un VAPA. Vérifiez que cette modélisation est correcte «localement».

Question 2.6 : modélisation du composant P

Mettez à jour si besoin est le comportement des caractéristiques du pont assurées par le composant P. Vérifiez que cette modélisation est correcte «localement».

Question 2.7 : modélisation du composant CTRLP

Mettez à jour si besoin est le comportement des caractéristiques du CTRLP. Vérifiez que cette modélisation est correcte «localement».

Question 2.8 : assemblage

Procédez à l'assemblage des composants que vous avez modélisé. Vérifiez que cet assemblage est cohérent et que les «propriétés triviales attendues» sont bien présentes dans cette spécification.

Question 2.9 : vérification

Procédez à la vérification formelle du modèle pour les propriétés suivantes :

P'_1 : Il n'y a pas de collision (i.e. deux véhicules circulants en sens inverse) sur le pont.

P'_2 : Un véhicule normal qui arrive est certain de passer sur le pont à l'issue d'une durée bornée.

P_3 : Un véhicule prioritaire qui arrive est certain de passer sur le pont à l'issue d'une durée bornée avant tout autre véhicule normal qui n'est pas encore entré sur le pont.

Vous exprimerez ces propriétés en des termes vérifiables et utiliserez les outils offerts par CPN-AMI pour démontrer leur véracité. Vous vous intéresserez à des valeurs remarquables des paramètres identifiés pour le système : N_{VAA}, N_{VAB}, $Capa_p$ et N_{alt} (en justifiant le choix des rapports entre ces valeurs).

Question 2.10 : retour sur les spécifications du système

Vous proposerez une solution cohérente en termes d'algorithmique répartie pour chacun des composants identifiés dans le système à des fin d'implémentation par une équipe de développeurs. Identifiez les propriétés à préserver et surtout la façon dont vous suggérez qu'elles devront être validées sur le système réel.

3 Structure de la livraison

Vous trouverez dans l'archive contenant le sujet, des fichiers /LaTeX contenant un modèle type de rapport. Vous pouvez utiliser un autre traitement de texte que /LaTeX pour rédiger ce rapport mais vous **devez** respecter la structure qui vous est proposée. ***La non observation de ces consigne entraînera des pénalitées***.

Pour nous rendre votre travail, vous ferez une image disque au sens MacOS du terme. Votre image disque devra respecter le format suivant (cf Figure 4). Le rapport sera à la racine et comprendra les nom des deux étudiants du binôme. Les fichiers sources de ce rapport seront localisé dans le répertoire `sources-rapport` qui contiendra également un `Makefile` permettant de produire le rapport sans se poser de questions.

Les répertoires associés à chaque question contiendront les fichiers de modèle :

– si vous utilisez Macao vous mettrez les fichiers Macao (postfixés par `.mco`),
– si vous utilisez Coloane, vous mettrez le résultat d'une exportation Coloane (fichiers postfixés par `.model`).

Pour chaque formule de logique temporelle que vous souhaitez vérifiez, un fichier `<formule>.txt` qui contiendra le texte de la formule à insérer dans l'outil Prod. Bien sûr, la formule sera référencée clairement dans le rapport (avec le nom du fichier associé).

le non-respect de ces conventions risque d'indisposer vos correcteurs en leur compliquant la tâche ;-). Par ailleurs, sachez que, indépendamment des techniques utilisées, c'est la façon de travailler que toute entreprise ou équipe digne de ce nom attendra de vous (souvenez-vous en pour le stage ;-).

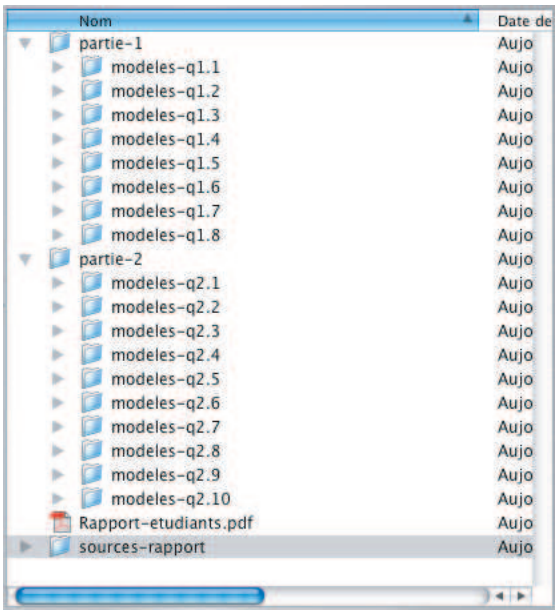


FIG. 4 – Structure de l'image disque que vous devez rendre.

Vous déposerez ce dmg dans un répertoire sur votre compte unix. Vous spécifierez ensuite le chemin absolu de ce répertoire en soumettant votre travail avec delivery_builder.

Annexe : éléments de syntaxe des formules dans Prod

PROD in CPN-AMI formula structure	
query [verbose] [node] formula	If **node** is set, the formula is evaluated on all reachability graph's nodes. Otherwise, it is evaluated on the initial state ondly ? If **verbose** is set, the verbose mode will be actrivated. Important : all nodes in the reachability graph are numbered. You can use these identifiers to point out some nodes.

Atomic formulas	
PlaceName1 >= Placename2	PlaceName1 contains less tokens than PlaceName2.
PlaceName != <..>	PlaceName marking is different from one signle non colored token.
Placename == 2<.1,2.> + <.1,3.>	PlaceName marking is equal to one composed colored token <1,3> plus two composed colored tokens <1,2>.
card (PlaceName) > 1	The number of tokens in PlaceName is greater than 1.
card(PlaceName :(field[0]==2)) >= 1	The number of composed tokens in place PlaceName for wich the first field is "2" is greater or equal to 1 (be aware that field numbering starts from 0).
and/or/not	Usual logical operators

Temporal formulas	
AX (formula)	Next on all branches.
EX (formula)	Next on some branch.
AG (formula)	Henceforth on all branches.
EG (formula)	Henceforth on some branch.
AF (formula)	Eventually on all branches.
EF (formula)	Eventually on some branch.
AU (formula, formula)	Until on all branches.
EU (formula, formula)	Until on some branch.
implies ($formula_1$, $formula_2$)	$formula_1$ implies $formula_2$.