

*Projet SF*

---

# *RAPPORT*

---

Modélisation et Vérification du comportement de véhicules  
automatiques sur un pont à voie unique

Réalisé par *DOAN Cao Sang*

5 décembre 2015

# Chapitre 1

## Propriété

*L*e programme doit satisfaire les propriétés suivantes :

1. Il n'y a pas de collision (i.e. deux véhicules circulants en sens inverse) sur le pont.
2. Un véhicule qui arrive est certain de passer sur le pont à l'issue d'une durée bornée.

## Chapitre 2

# Explication

### 2.1 Question 1.1

#### 2.1.1 Vaa

**VerificationA** est une place, qui lie le composant *Vaa* avec le contrôleur *CTRLP*, lors que une véhicule A veut traverser le pont, d'abord elle communique avec le contrôleur *CTRLP* pour avoir son autorisation.

**OKControlleurA** est une place qui lie entre *Vaa* et *CTRLP*, lors que *Vaa* peut traverser sur le pont, *CTRLP* met son jeton pour que *Vaa* puisse passer.

**Pause** est une place

Lors que un interlocuteur a acquiert le bus principal, il envoie un message au l'autre, soit le message arrive au destinataire, soit le message se tombe dans un puits. Dans le 1<sup>er</sup> cas, l'émetteur reçoit un ACK au bout un certain temps arbitraire dès que le récepteur reçoit le message mais fini. Dans le 2<sup>ème</sup> cas, si l'émetteur détecte la perte, il renvoie le message jusqu'à quand il est sûr qu'il n'y a pas de perte de message.

Selon les invariant ci-dessous, on peut trouver les invariants qui assurent les propriétés demandées sont :

- *Invariant 7* : Le cable principal soit occupé, soit libre, donc qu'il ne peut jmais à la fois libre et occupé.
- *Invariant 1* : A un instant quelconque, il n'y a qu'un seul message( MSG ou ACK) en transit.
- *Invariant 3* : Deux interlocuteurs sont soit dans l'état d'écoute le MSG et envoie ACK quand il reçoit le MSG, soit dans l'état d'émission du MSG, si le MSG est perdu, il doit renvoyer au récepteur.
- *Invariant 5* : Cet invariant assure que la 1<sup>ère</sup> propriété a été validé, il y a toujours un ACK suivi du MSG.

```
query node ((card(DansPontA)==1) and (card(DansPontB)==1))
```

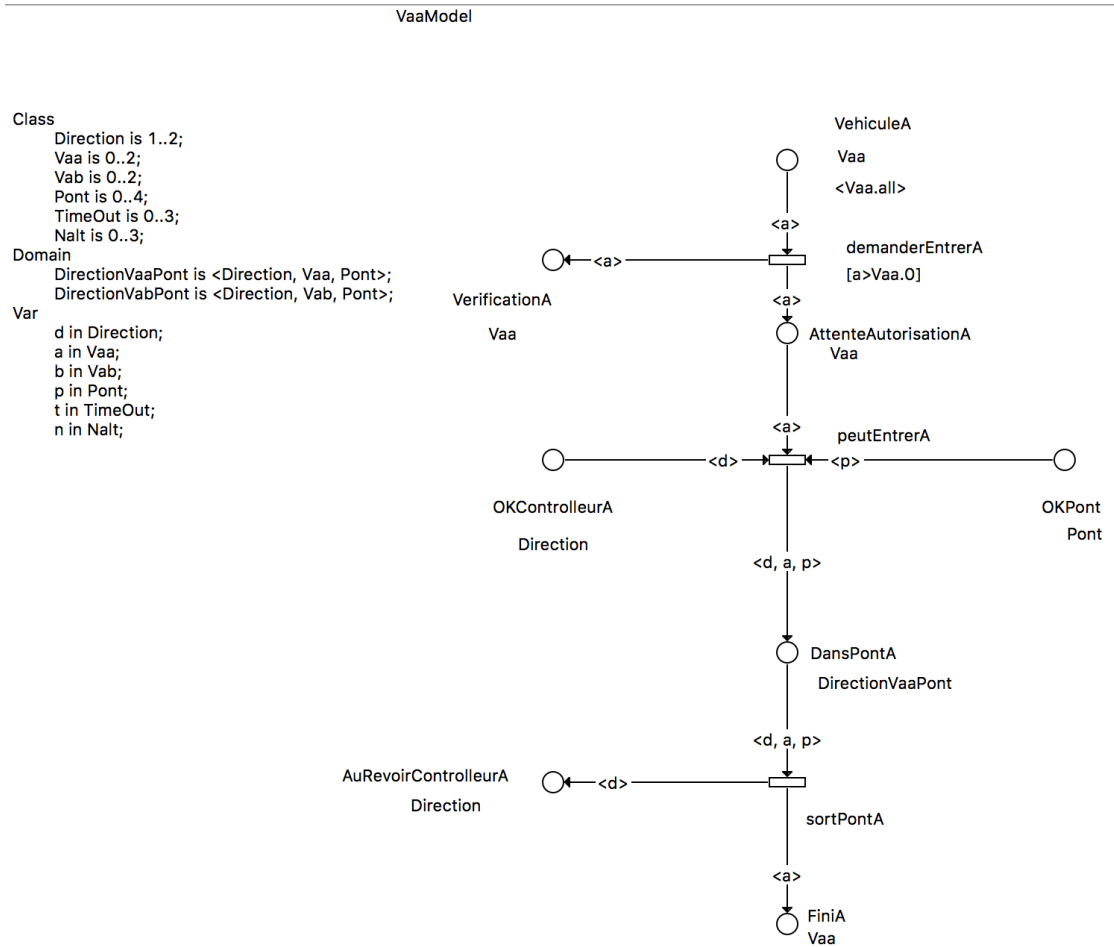


FIGURE 2.1 – Le composant véhicule automatisée A.

```
=====
This service relies on the prod model checker developped
at the Helsinki University of technology (laboratory of
computer science, http://www.tcs.hut.fi/Software/prod)
=====
```

Integration in CosyVerif by F. Kordon (UPMC, 2013)

Installing Prod

Installing gph2dot

Installing model data for persistent storage

PN syntax checker by X. Bonnaire 1992-1996, A. Diagne 1996-1997, I. Mounier 1993-2000

Statistics for this net

Number of places 22

Number of Queues 0

Number of Transitions 13

Number of arcs 55

Generating prod format

Generating and compiling code for the model

Generating the state space (this may take a while;-)

---

Statistics about the reachability graph provided by prod are shown below

Number of nodes: 283688

Number of arrows: 1033700

Number of terminal nodes: 0

Number of nodes that have been completely processed: 283688

Number of strongly connected components: 229305

Number of nontrivial terminal strongly connected components: 1

Evaluating the provided CTL formula

0 paths

Built set %1

```
=====
Service completed OK
=====
```

Debug information

```
=====
CosyVerif directory -> /tmp/alligator-5454896919309414835
```

```
Persosistent directory -> /tmp/ProdInCosyVerif_f369dd251b6b1eac4a8afd256ef6d33b
=====
```

```
Execution trace follows below
```

```
+ '[' '!' -f prod_model ']'
+ echo 'PN syntax checker by X. Bonnaire 1992-1996, A. Diagne 1996-1997, I. Mounier
+ /tmp/ProdInCosyVerif_f369dd251b6b1eac4a8afd256ef6d33b/FkSandrine -s -i prod_876380
+ '[' '!' -f .stats ']'
++ wc -c
++ grep colores .qualif
+ CLASS=8
+ sed -e s/://
+ cat .stats
+ echo 'Generating prod format'
+ /tmp/ProdInCosyVerif_f369dd251b6b1eac4a8afd256ef6d33b/matrice -s -i camix_model -m
++ tr -s ' '
++ cut -d ' ' -f 2
++ wc -l rdp_matrix
+ CHECK=rdp_matrix
+ '[' rdp_matrix = 0 ']'
++ grep NAME_NOT_IDENTIFIER matrice_output
+ '[' ' ' ']'
+ '[' -f prod_model.gph ']'
+ echo 'Generating and compiling code for the model'
+ prod prod_model.init
+ '[' -f prod_model ']'
+ echo 'Generating the state space (this may take a while;-)''
+ ./prod_model -m 1024000
```

```
./prod_model.aws is used directly
```

```
./prod_model.gph is used directly
```

```
./prod_model.adr is used directly
```

```
+ echo statistics
+ strong prod_model
+ sed -e s/./#/g
+ probe prod_model
+ echo ---
+ echo 'Statistics about the reachability graph provided by prod are shown below'
+ cat statistics.data
+ echo 'Evaluating the provided CTL formula'
+ cat ctl.prb prod_801592428106530532.ctlprod
+ sed -e 's/.*#/g'
+ probe prod_model
```

```
+ cat prod_formula_result
```