Simuler k canaux avec un seul

Verifying Lossy Channel Systems has Nonprimitive Recursive Complexity de Philippe Schnoebelen ([Sch01])

Emmanuel Cornet
École normale supérieure

Cours de Vérification – MPRI

16 février 2005

Plan

Présentation générale

- « La vérification des systèmes à canaux non fiables est non primitive récursive » : l'accessibilité et la terminaison sont des problèmes de complexité non primitive récursive.
- Réduction de k canaux à un seul : montre que le résultat s'étend aux système à un seul canal.

Le principe

On veut simuler un système à $k \geq 1$ canaux avec un système à **un seul canal**.

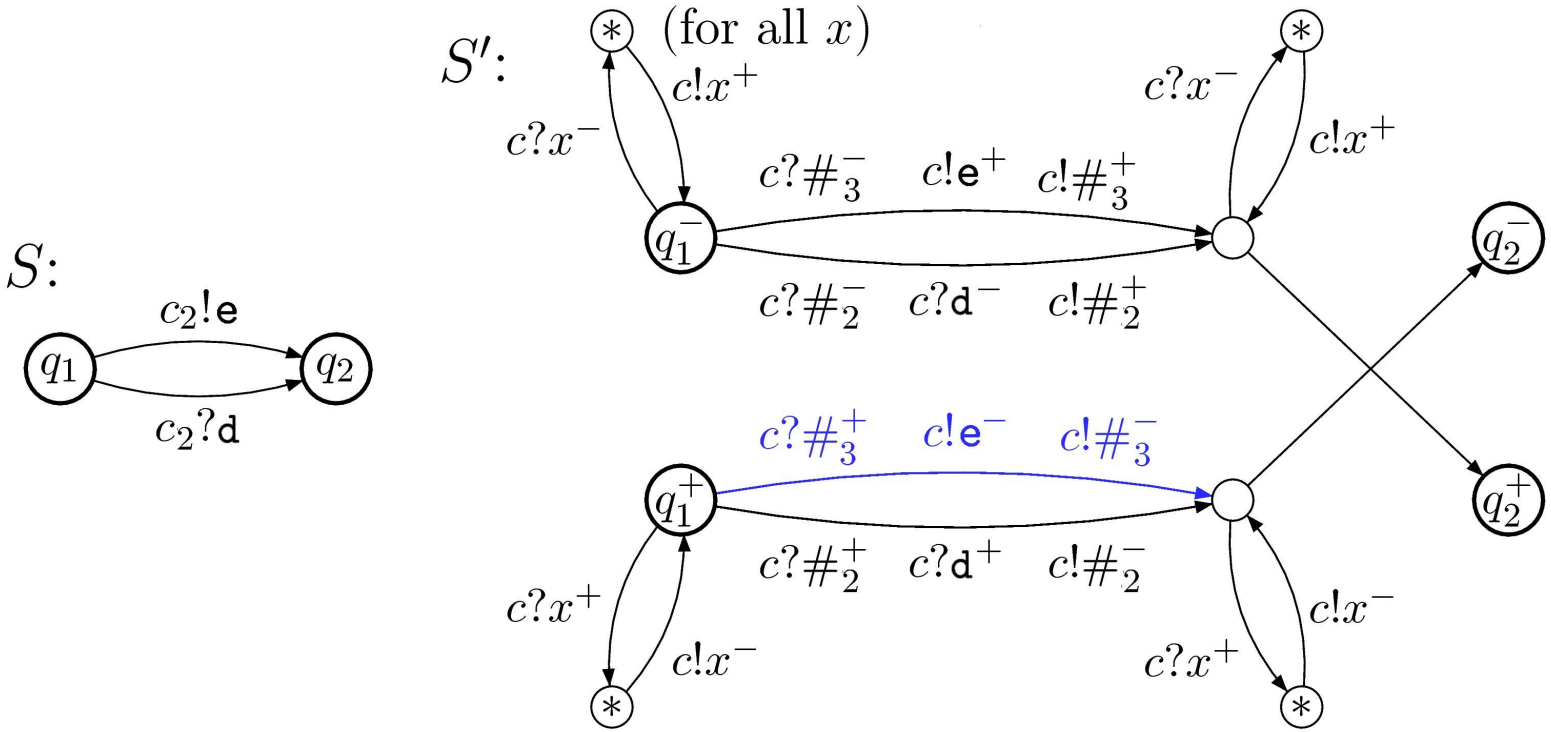
- On suppose (sans perte de généralité) que les k alphabets sont disjoints
- On rajoute k symboles dans l'alphabet : $\#_1, \ldots, \#_k$
- On utilise deux copies du même alphabet, de polarité $+$ ou $-$
- On écrit les k mots à la suite dans l'unique canal, séparés par des « marqueurs » $\#_i$

Exemple : codage de $(\mathtt{ab}, \mathtt{dc}, \mathtt{ggf})$

$$\#_1^+ \mathtt{a}^+ \mathtt{b}^+ \#_2^+ \mathtt{d}^+ \mathtt{c}^+ \#_3^+ \mathtt{g}^+ \mathtt{g}^+ \mathtt{f}^+$$

La construction – écriture

(Version corrigée)



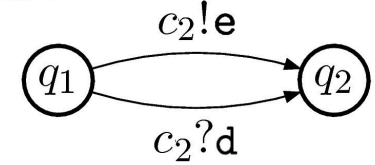
$\#_{\mathbf{1}}^+\mathtt{a}^+\mathtt{b}^+\#_2^+\mathtt{d}^+\mathtt{c}^+\#_3^+\mathtt{g}^+\mathtt{g}^+\mathtt{f}^+$

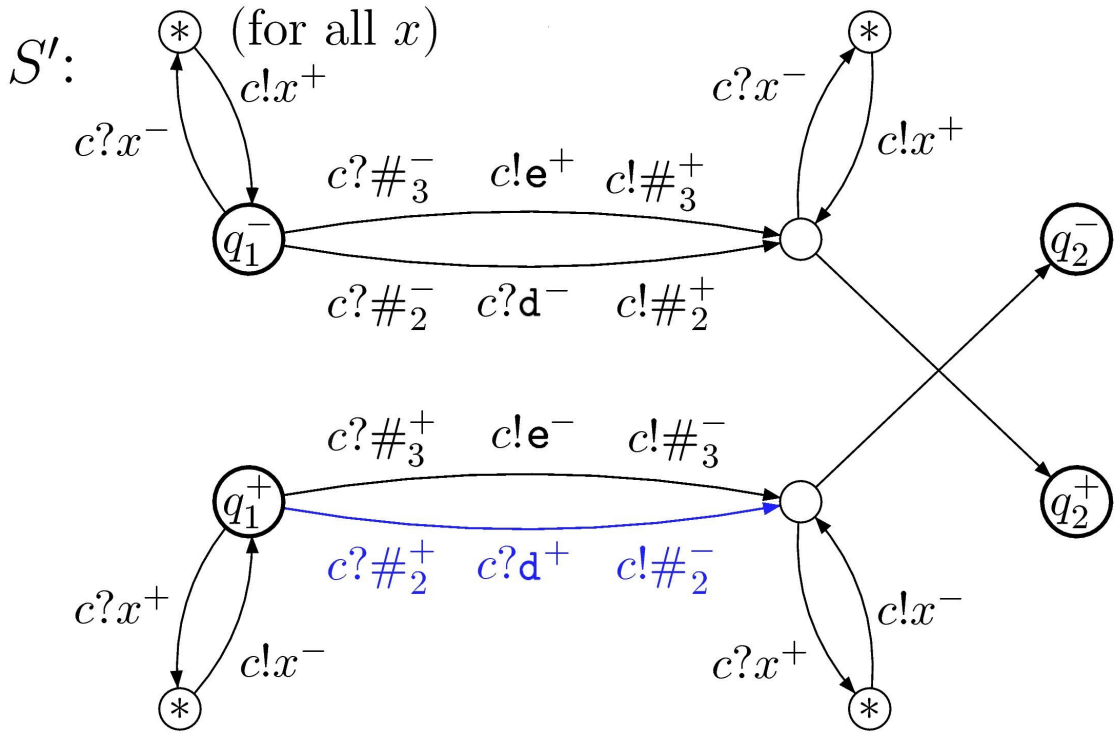
$\longrightarrow$ $\#_{\mathbf{3}}^+\mathtt{g}^+\mathtt{g}^+\mathtt{f}^+\#_{\mathbf{1}}^-\mathtt{a}^-\mathtt{b}^-\#_2^-\mathtt{d}^-\mathtt{c}^-$

$\longrightarrow$ $\mathtt{g}^+\mathtt{g}^+\mathtt{f}^+\#_1^-\mathtt{a}^-\mathtt{b}^-\#_2^-\mathtt{d}^-\mathtt{c}^-\mathtt{e}^-\#_{\mathbf{3}}^-$

$\longrightarrow$ $\#_{\mathbf{1}}^-\mathtt{a}^-\mathtt{b}^-\#_2^-\mathtt{d}^-\mathtt{c}^-\mathtt{e}^-\#_3^-\mathtt{g}^-\mathtt{g}^-\mathtt{f}^-$

La construction – lecture





$\#_{\mathbf{1}}^+\mathtt{a}^+\mathtt{b}^+\#_2^+\mathtt{d}^+\mathtt{c}^+\#_3^+\mathtt{g}^+\mathtt{g}^+\mathtt{f}^+$

$\longrightarrow$ $\#_{\mathbf{2}}^+\mathtt{d}^+\mathtt{c}^+\#_3^+\mathtt{g}^+\mathtt{g}^+\mathtt{f}^+\#_{\mathbf{1}}^-\mathtt{a}^-\mathtt{b}^-$

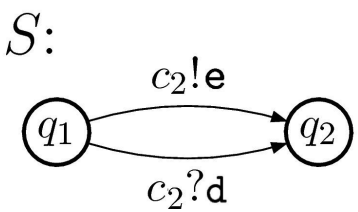
$\longrightarrow$ $\mathtt{c}^+\#_3^+\mathtt{g}^+\mathtt{g}^+\mathtt{f}^+\#_1^-\mathtt{a}^-\mathtt{b}^-\#_{\mathbf{2}}^-$

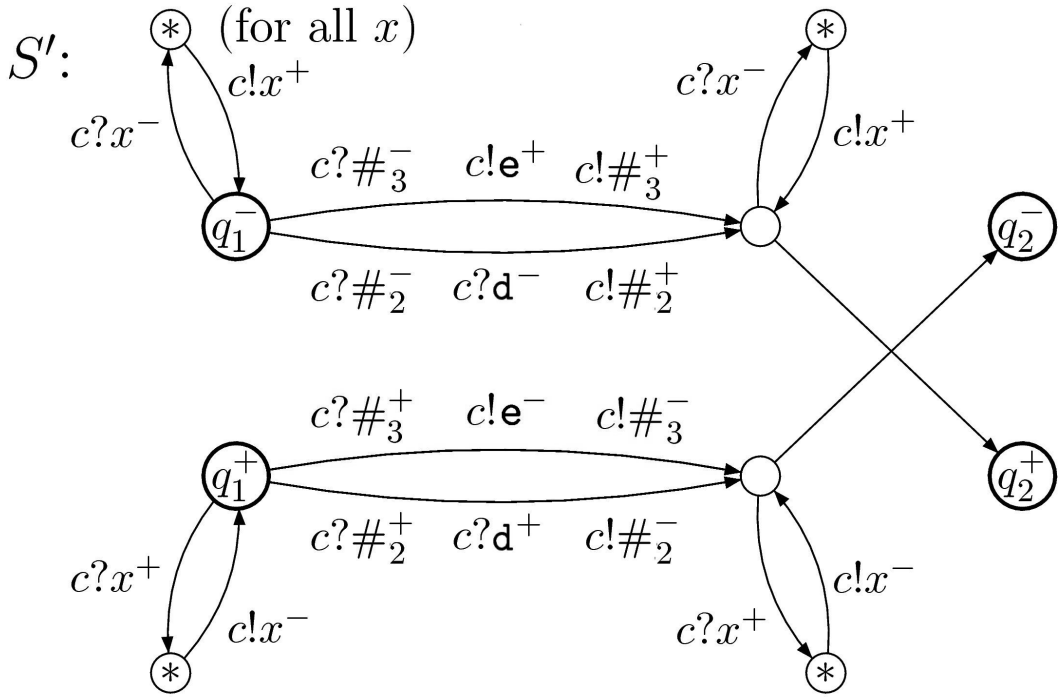
$\longrightarrow$ $\#_{\mathbf{1}}^-\mathtt{a}^-\mathtt{b}^-\#_2^-\mathtt{c}^-\#_3^-\mathtt{g}^-\mathtt{g}^-\mathtt{f}^-$

Une petite définition

Definition

On dira qu'une exécution de S' est **bien polarisée** si dans chaque état de contrôle q_i^α (sans considérer les états intermédiaires induits par les boucles), où $\alpha \in \{+; -\}$, le contenu du canal c de S' ne contient que des lettres de polarité α.





Une autre petite définition

Definition

On dira qu'une exécution $\langle q^\alpha, W^\alpha \rangle \xrightarrow{n} \langle q'^\beta, V^\beta \rangle$ de S' est **fidèle à** l'exécution sans pertes $\langle q, w_1, \ldots, w_k \rangle \xrightarrow{n} \langle q'', v_1, \ldots, v_k \rangle$ de S si :

- W est un codage de $\langle w_1, \ldots, w_k \rangle$;
- elle s'effectue sans pertes ;
- chaque transition de S est traduite dans S' selon la construction précédente ;
- elle est bien polarisée.

Deux choses peuvent rendre une exécution de S' non fidèle : la perte d'une ou plusieurs lettres, et le passage dans un nouvel état de contrôle avant d'avoir « évacué » les lettres de polarité précédente (exécution « mal polarisée »).

Un petit lemme pour la route

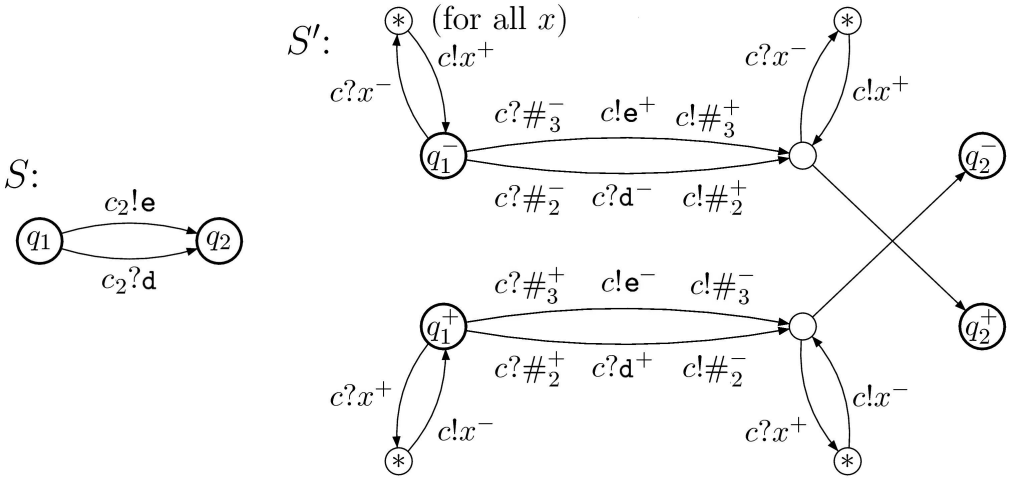
Lemma

Si l'exécution $\langle q^\alpha, W^\alpha \rangle \xrightarrow{n} \langle q'^\beta, V^\beta \rangle$ *de* S' *est fidèle à l'exécution* $\langle q, w_1, \ldots, w_k \rangle \xrightarrow{n} \langle q'', v_1, \ldots, v_k \rangle$ *de* S*, alors*

- $q' = q''$
- V^β *est un codage de* $\langle v_1, \ldots, v_k \rangle$

Démonstration.

Par induction sur la longueur de l'exécution. □



Accessibilité

Theorem

$$\sigma = \langle q, w_1, \ldots, w_k \rangle \xrightarrow{*} \langle q', v_1, \ldots, v_k \rangle = \sigma' \text{ dans } S$$

$$\Updownarrow$$

$$\rho = \langle q^\alpha, W^\alpha \rangle \xrightarrow{*} \langle q'^\beta, V^\beta \rangle = \rho' \text{ dans } S'$$

(où W et V sont les codages de $\langle w_1, \ldots, w_k \rangle$ et $\langle v_1, \ldots, v_k \rangle$)

Accessibilité – démonstration

Démonstration.

$\Downarrow$ Il existe une exécution dans S partant de σ et atteignant σ'. Si elle est sans pertes, d'après le lemme, il suffit de considérer l'exécution $\rho \xrightarrow{*} \rho'$ fidèle à $\sigma \xrightarrow{*} \sigma'$. Sinon, il faut considérer l'exécution qui « perd » les mêmes lettres aux mêmes moments.

$\Uparrow$ Il existe une exécution dans S' partant de ρ et atteignant ρ'.

- Si cette exécution est sans pertes et bien polarisée, alors elle est fidèle à une exécution $\sigma \xrightarrow{*} \sigma'$ de S et le résultat est vérifié.
- Si cette exécution est avec pertes, là encore on considère l'exécution de S qui perd les mêmes lettres aux mêmes moments.
- Si cette exécution n'est pas bien polarisée, alors la mauvaise polarisation apparaît nécessairement à la dernière transition (sinon le système se serait bloqué), et cela n'affecte pas l'accessibilité de ρ'. Il suffit alors de considérer l'exécution bien polarisée correspondante, fidèle à une exécution $\sigma \xrightarrow{*} \sigma'$ de S.

□

Terminaison

Theorem

$$S \text{ termine à partir de } \sigma_0 = \langle q, w_1, \ldots, w_k \rangle$$
$$\Updownarrow$$
$$S' \text{ termine à partir de } \rho_0 = \langle q^\alpha, W^\alpha \rangle$$

(où W est le codage de $\langle w_1, \ldots, w_k \rangle$)

Terminaison – démonstration

Démonstration.

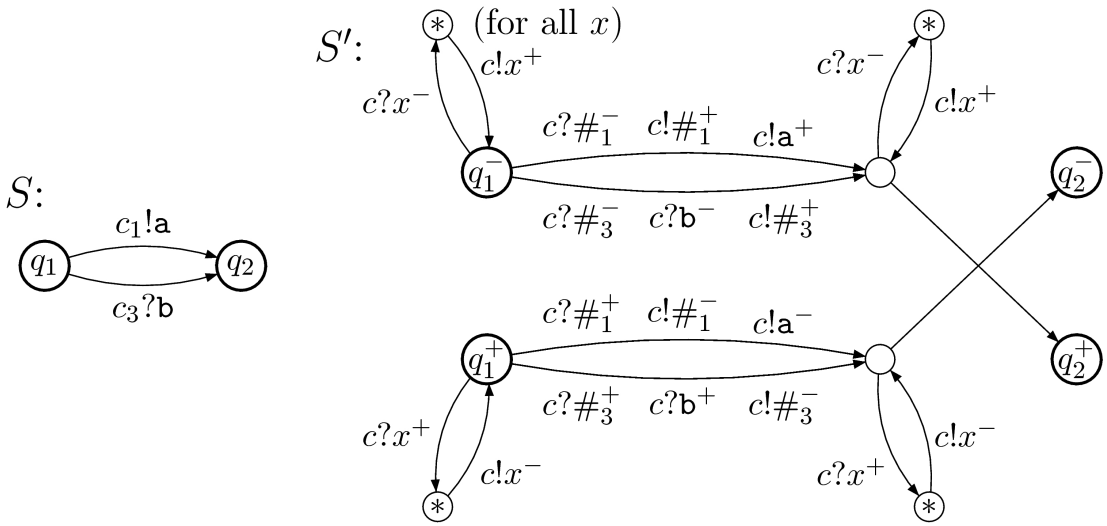
$\Downarrow$ S termine à partir de σ_0 ; soit une exécution de S' qui simule une exécution de S à partir de σ_0, et partant du codage de σ_0. Aucune des quatre boucles « $*$ » ne peut faire boucler le système car la polarité des lettres écrites est l'opposé de celle des lettres lues. Tous les autres « défauts » de simulation (pertes et mauvaise polarisation) ne peuvent que bloquer le système, et donc faire terminer l'exécution plus tôt. Par conséquent, l'exécution de S' termine.

$\Uparrow$ Montrons la contraposée : soit une exécution e de S, partant de σ_0, qui ne termine pas. Soit alors l'exécution e' de S' qui lui est fidèle : d'après le lemme, e' parcourt les mêmes états que e, et ne termine donc pas non plus.

□

Conclusion et compléments

- Un *bug* dans l'article, écriture et lecture en tête :



- Abdulla–Jonsson ([AJ96]) : préservation de l'**accessibilité** d'un état de contrôle **infiniment souvent**, mais pas de l'**acessibilité** ni de la **terminaison** (disjointure des alphabets nécessaire)

Bibliographie

- L'article étudié :

  [Sch01] Philippe Schnoebelen, Verifying Lossy Channel Systems has Nonprimitive Recursive Complexity, *Information Processing Letters* 83(5), pages 251-261, 2002.

- La construction précédente d'Adbulla et Jonsson :

  [AJ96] Parosh A. Abdulla et Bengt Jonsson, Undecidable verification problems for programs with unreliable channels, *Information and Computation*, 130(1) :71–90, 1996.

Retrouvez les documents sur le Web

Cette présentation est disponible au format PDF à l'adresse

`www.eleves.ens.fr/home/cornet/Canaux_presentation.pdf`