

Options RTL en date du 28/1/97

Certaines de ces options n'ont pas d'intérêt pour l'utilisateur

Debug général

- d39** est la prochaine option libre
- d1** montre les inférences de la SOS LOTOS (ne marche pas très bien)
- d2** montre le nombre d'états LOTOS
- d3** montre les arcs rencontrés entre états LOTOS
- d4** montre les états LOTOS complets (à utiliser avec -d2)
- d25** détaille plus chaque noeud de l'arbre LOTOS
- d22** montre les ressources pour chaque -d2
- d23** pour analyser les collisions dans les ensembles contenant les états engendrés
- id1** pour analyser les collisions dans n'importe quel ensemble (utilisé comme -d23)
- d35** montre l'état qui doit être développé
- d37** ne montre pas les erreurs et n'aborte pas
- d38** montre les erreurs (warnings) et n'aborte pas

Options de vérification

- lotos -verif** construit le graphe LOTOS
- verif** construit le graphe avec temps discret, cf options de simulation
- TG1** crée uniquement le graphe temporisé

Deux moteurs de génération :

- BEH-TG** ancien, comme détaillé dans article FORTE'95
- BEH-DTA** option par défaut

-**shownm**

-**d19** montre (n,m) de beh même si horloge non valide

-**d24** montre le résultat de make_list_std_clock

Ensemble U :

-**OLDURG** gestion des arcs faibles à la place de U

-**NEWURG** option par défaut

-**MAXCLOCK** option nécessaire à KRONOS

-**TG2** réalisation de la minimisation à la volée

-**TG3** génération du DTA puis minimisation

-**cray** (-dosubgraph -doregular)

- d11** montre l'information propre à -cray
- d12** détaille le fonctionnement de -cray
- d20** montre le calcul de its_size()
- d29 -d30 -d31 -d32 -d33** montrent les états de manière plus compacte (indépendants de -d11)
 - d29** montre les listes d'ids de let de ISCH avec ses valeurs
 - d30** omission de l'expression RT-LOTOS dans ISCH
 - d31** montre une liste énumérée de chaque paramètre de ISREG
 - d32** l'expression RT-LOTOS de ISREG n'est pas montrée
 - d33** montre le nombre de CHREG
 - d34** omission de l'expression RT-LOTOS dans CHREG

Options de simulation

Par défaut, construction d'un arc temporel égal à TMAX \neq 0 pour chaque état

- TRAND** construit un arc entre 1 et TMAX
- TDETN** construit arcs N, 2*N, 3*N, ..., TMAX
- SIM0** (par défaut) donne la priorité aux actions non temporelles
- SIM-1** exécute de manière aléatoire une action temporelle ou non temporelle
- d18** montre le résultat de l'appel de rand()
- seed-N** détermine le seed de rand()
- d26** montre les actions sensibilisées dans un choix non déterministe
- d27** montre la valeur de TMAX
- d28** montre valrand,prerand

Debug de verification en temps reel

- d5 -d6 -d7 -d8 -d9 -d13 -d14 -d10 -d21**
- d5** montre la construction de la partition initiale de chaque état
- d6** montre les pas de l'algorithme de stabilisation repères dans le fichier produit :
 - génération des successeurs d'un état du DTA
 - State
 - quand un état du DTA devient accessible
 - Adding DTA state
 - Making node
 - Adding state partition
 - Arc (N
 - fin du step 1 de l'algo
 - Memory =
 - END step 1
 - step 2 de l'algo
 - u (level N in step 2X)=
 - New node y
 - New node z

```

    fin de la stabilisation
    Memory =
    END in level
    les arcs du graphe de classes stables
    ( N
-d7 -d8 test de propriétés des opérations de l'algorithme
-d9 montre le calcul de région stable
-d13 engendre .aut avec tout le graphe à partir de forest066 : :fait_atg()
-d14 test les propriétés de l'algorithme
-d15 réintroduit un bug ancien dans le calcul de inv_int_t
-d10 (obsolète) gérait les phases importantes de la minimisation sortie
    standard et fichiers .fc2/.aut :
    B indique la configuration analysée dans la phase 1
    C indique la partition nécessaire (un niveau supplémentaire)
    D indique que la partition a été testée et est stable
-d21 (obsolète) montrait le chemin suivi pour les configurations
-d16 -d17 debug de java :
-d16 montre le calcul de value_expr
-d17 montre les opérations avec les objets Java
-max-real-t-N
-max-spec-t-N
-debug-spec-t-N
-debug-real-t-N
-max-mem-size-N
-max-mem-rss-N
-max-state-N
-max-symb-state-N
-max-recursion-N utilisé dans expand pour détecter un processus "non-gardé"
-lts-size-N
-lts-collision-size-N
-symb-lts-size-N
-symb-lts-collision-size-N
-ATG engendre les graphes *.fc2
-AUT engendre les graphes *.aut
    *.lot.tgN.[ren|fc2|aut] contient des DTAs
-d36 engendre le DTA accessible au format *.lot.rtgN.[ren|fc2|aut]
    algorithme de minimisation :
    *.lot.rgN.[ren|fc2|aut] contient les régions du dernier niveau
    *.lot.rfN.[fc2|aut] contient les régions dans chaque niveau
    pour utiliser bcg_edit qui charge les fichier .ps :
    - voir le script aut2bcg.pl
    - manuellement, on peut faire :

```

```
bcg_io f.aut f.bcg  
bcg_draw -ps f.bcg
```

~/bin/atg contient un pointeur vers Autograph

Utiliser "Save atg as" pour sauver un graphe développé et continuer le
jour suivant.