

ETVO: librairie C++ pour la description des Graphes d'Événements Temporisés Valués

MSR'19 Angers

B.Cottenceau, L.Hardouin, J.Trunk
LARIS Angers, TU Berlin

Novembre 2019

ETVO=(Event|Time)-Variant Operators

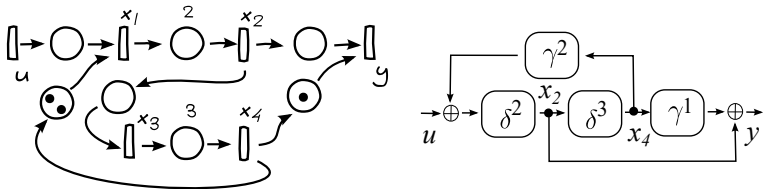
La librairie ETVO gère des expressions rationnelles constituées sur un ensemble d'opérateurs $\{\gamma^n, \delta^t, \mu_m, \beta_b\}$.

Ces opérateurs décrivent des phénomènes dynamiques relevant de la classe des systèmes $(\max, +)$ cycliques.

- ETVO étend la librairie MinMaxGD [Hardouin(06)]
- ETVO est l'implémentation de
 - (1) Cottencaeu B., Hardouin L. et Le Corrond E., "Représentation Tridimensionnelle de la Dynamique des Graphes d'Événements Temporisés Généralisés", MSR'09
 - (2) Cottencaeu B., Hardouin L., Boimond J.-L., "Modeling and Control of Weight-Balanced Timed Event Graphs in Dioids", IEEE TAC, 2014
 - (3) Trunk J., "On the modeling and control of extended Timed Event Graphs in dioids", thèse U.Angers/TU Berlin, Nov. 2019

Transfert de Graphes d'Événements Temporisés (GET)

Opérateurs : γ^n (décalage dans la numérotation) δ^t (décalage temporel)



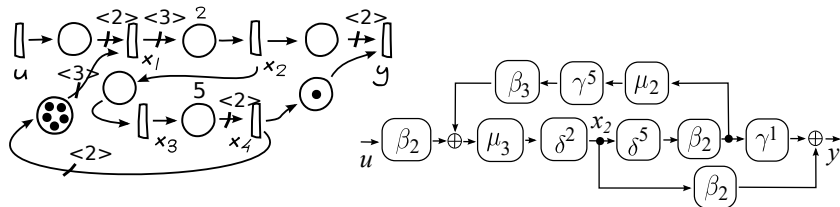
Dynamique : $\{x = Ax \oplus Bu, y = Cx\}$. Le **transfert** est $y = CA^*Bu$.

$$A = \begin{pmatrix} \cdot & \cdot & \cdot & \gamma^2 \\ \delta^2 & \cdot & \cdot & \cdot \\ \cdot & e & \cdot & \cdot \\ \cdot & \cdot & \delta^3 & \cdot \end{pmatrix}, B = \begin{pmatrix} e \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}, C = (\cdot \quad e \quad \cdot \quad \gamma^1).$$

$$y = CA^*Bu = (\delta^2 \oplus \gamma^1 \delta^5)(\gamma^2 \delta^5)^* u.$$

GET valués: opérateurs élémentaires $\delta^\tau, \gamma^\nu, \mu_m, \beta_b$

μ_m : duplication d'événements β_b :regroupement d'événements



$$A = \begin{pmatrix} \cdot & \cdot & \cdot & \beta_3 \gamma^5 \mu_2 \\ \delta^2 \mu_3 & \cdot & \cdot & \cdot \\ \cdot & e & \cdot & \cdot \\ \cdot & \cdot & \beta_2 \delta^5 & \cdot \end{pmatrix}, B = \begin{pmatrix} \beta_2 \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}, C = (\cdot \quad \beta_2 \quad \cdot \quad \gamma^1).$$

Le transfert $H = CA^*B$ s'écrit sous une forme $H = p \oplus q(\gamma^\nu \delta^\tau)^*$.

$$A = \begin{pmatrix} \cdot & \cdot & \cdot & \beta_3 \gamma^5 \mu_2 \\ \delta^2 \mu_3 & \cdot & \cdot & \cdot \\ \cdot & e & \cdot & \cdot \\ \cdot & \cdot & \beta_2 \delta^5 & \cdot \end{pmatrix}, B = \begin{pmatrix} \beta_2 \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}, C = (\cdot \quad \beta_2 \quad \cdot \quad \gamma^1).$$

```

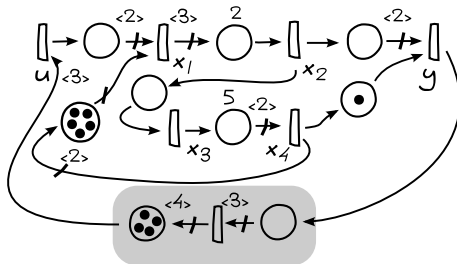
#include "etvo.h"
using namespace etvo;
int main(){
    matrix<seriesEd> A(4,4), B(4,1), C(1,4);
    B(0,0)=eb(2); C(0,1)=eb(2); C(0,3)=eg(1);
    A(0,3)=eb(3)*eg(5)*em(2); A(1,0)=em(3)*ed(2);
    A(2,1)=eg(0); A(3,2)=eb(2)*ed(5);
    matrix<seriesEd> H=C*A.star()*B;
    H(0,0).toRight();
    std::cout << H(0,0) << "\n";
    //((m3.b4.g2+g1.m3.b4).d2+(g1.m3.b4.g2+g2.m3.b4).d7).[g2.d7]*
}

```

Synthèse de contrôleurs (régulateurs)

Feedback neutre: retarde l'entrée au plus tard, sans changer la sortie.

$$F_{opt} = H \setminus H \phi H$$



La librairie ETVO implémente les calculs de résiduation \setminus, ϕ .

$F = \gamma^4 \mu_4 \beta_3 \preceq F_{opt}$ est neutre (non optimal)

$$F_{opt} = (\gamma^2 \delta^7)^* (\gamma^2 \mu_4 \beta_3 \gamma^2 \oplus \gamma^4 \mu_4 \beta_3 \oplus (\gamma^2 \mu_4 \beta_3 \gamma^2 \oplus \gamma^4 \mu_4 \beta_3 \gamma^1) \delta^5)$$