

Schéma-blocs avec PGF/TIKZ

Papanicola Robert

21 janvier 2020

version 1.8 ajout des commandes permettant de tracer des comparateurs/sommeurs dans les boucles de retour

version 1.62 retour de la boucle foreach

version 1.61 ajout de la commande sbStyleSum afin de définir le style graphique des comparateurs et sommateurs.

version 1.6 Remplacement de la boucle foreach de tikz par la boucle `\newforeach` du package loops.

version 1.5.1 Un bug dans la fonction `remember` de `foreach` impose de choisir au minimum la version 2.1 cvs de `pgf`.

version 1.5 Modification du dessin des comparateurs et sommateurs (ajout des symboles avec l'opérateur à l'extérieur). Ajout des commandes de tracé de chaînes de blocs et de boucles. les vieilles définitions ont été désactivées (mise en commentaires).

version 1.2 redéfinition des noms de commandes, ajout du préfixe `sb`, quelques nouvelles commandes (`\sbBlocSeu` ..)

version 1 version initiale

1 Schéma-blocs avec PGF/TIKZ

Les macros suivantes permettent de faciliter le dessin de schéma-blocs (block-diagram), elles s'appuient sur le package `pgf` et les macros `tikz`.

Les macros ont été adaptées à partir des exemples d'utilisation de la librairie `pgf` de Till Tantau et `TikZ` de Kjell Magne Fauske et principalement `block-diagram`.

1.1 Utilisation typique

L'utilisation typique de cet ensemble de macro-commandes est produire des diagrammes fonctionnels tel celui présenté sur la figure suivante.

The typical use of this set of macro-commands is to produce functional diagrams such that presented on the following figure.

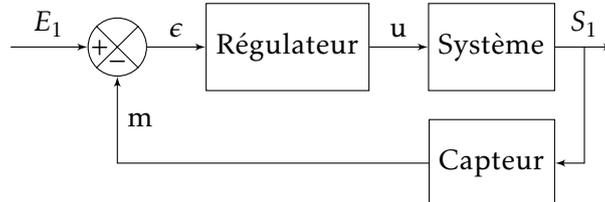


FIGURE 1 – Utilisation typique

On retrouve dans ces schémas les principales fonctionnalités des macros	<code>\begin{tikzpicture}</code>
— des blocs :	<code>\sbEntree{E}</code>
— de la gauche vers la droite,	<code>\sbComp{comp}{E}</code>
— de la droite vers la gauche ;	<code>\sbRelier[E_1]{E}{comp}</code>
— un comparateur ;	<code>\sbBloc{reg}{Régulateur}{comp}</code>
— une entrée ;	<code>\sbRelier[ϵ]{comp}{reg}</code>
— une sortie ;	<code>\sbBloc{sys}{Système}{reg}</code>
— des liens :	<code>\sbSortie{S}{sys}</code>
— simple entre blocs alignés,	<code>\sbRelier[S_1]{sys}{S}</code>
— entre un lien et un bloc ,	<code>\sbDecaleNoeud[4]{S}{U}</code>
— entre deux blocs (ou comparateur) non alignés,	<code>\sbBlocr{cap}{Capteur}{U}</code>
— un retour direct.	<code>\sbRelieryx{sys-S}{cap}</code>
Le code utilisé pour décrire le schéma est précisé ci-contre.	<code>\sbRelierxy[m]{cap}{comp}</code>
	<code>\end{tikzpicture}</code>

1.2 Environnement

Les macros suivantes s'utilisent dans l'environnement `tikz-picture`.
Nota : vous devez avoir installé la dernière version du package `pgf!`

1.3 Entrée et noeud

Tous les blocs sont dessinés en relatif par rapport à un noeud d'entrée, la construction du schéma ne peut donc débuter qu'après avoir défini le premier noeud avec la commande `\sbEntree{nom}`.
 Les commandes `\sbDecaleNoeudx[distance]{N1}{N2}` et `\sbDecaleNoeudy[distance]{N1}{N2}` permettent de positionner un nouveau noeud {N2} par rapport au noeud précédent {N1} , respectivement

- suivant x - horizontalement de la gauche vers la droite
- suivant y - verticalement du haut vers le bas de la page

Cette commande est nécessaire pour démarrer un nouvelle branche ou pour positionner une nouvelle entrée.

La distance optionnelle [distance] doit être précisée sans unité et est comptée en em, la valeur par défaut est de 5em.

Remarque : il est toujours possible de positionner les différents noeuds en absolu dans la page en utilisant les commandes spécifiques de tikz (`\node...`).

1.4 Bloc

1.4.1 Utilisation

deux commandes principale permettent d'obtenir le dessin d'un bloc fonctionnel
 — la première permet le dessin d'un bloc seul ;

`\sbBloc[distance]{nom}{contenu}{bloc precedent}`

— la seconde permet le dessin du bloc et du lien avec le bloc précédent

`\sbBlocL[distance]{nom}{contenu}{bloc precedent}`

avec les paramètres suivant :

distance ce paramètre optionnel permet de positionner le bloc par rapport au bloc précédent (la valeur par défaut est 2 em), cette distance est l'intervalle entre les deux blocs ;

nom ce paramètre permet de nommer le noeud associé au bloc, pour faire référence à ce bloc, il faudra utiliser ce paramètre (pour relier les blocs) ;

contenu ce paramètre précise le contenu du bloc, cela peut être aussi bien du texte qu'une fonction mathématique comme $\frac{K_c}{1 + \tau \cdot p}$ en tapant `\dfrac{K_c}{1 + \tau \cdot p}` (ne pas oublier les \$) ;

bloc precedent ce paramètre permet de préciser le nom du bloc précédent, chaque bloc est positionné relativement au bloc précédent avec la distance **distance**.

1.4.2 Exemple

```
\begin{tikzpicture}
\sbEntree{E}
\sbBloc{bloc1}{contenu}{E}
\sbBloc{bloc2}{

$$K_c \frac{1 + \tau \cdot p}{1 + \frac{2 \cdot z}{\omega_n} p + \frac{p^2}{\omega_n^2}}$$

}{bloc1}
\sbBlocL[4]{bloc3}{Bloc lié}{bloc2}
\end{tikzpicture}
```

- on notera la présence de `\entree` pour positionner le premier bloc;
- le nom de chaque bloc est unique;
- le deuxième bloc s'est adapté en hauteur et largeur en fonction du contenu;
- le troisième bloc est décalé de 4em du précédent et relié.

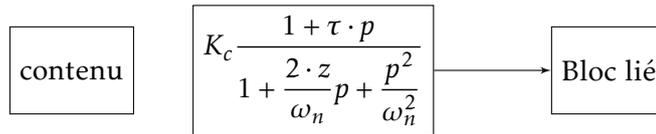


FIGURE 2 – commandes `\sbBloc` et `\sbBlocL`

1.4.3 Autres commandes de Bloc

Bloc de la chaîne de retour La commande `\sbBlocr...` (respectivement `\sbBlocrL...`) permet de tracer les blocs de la chaîne de retour de la droite vers la gauche. les paramètres de commande sont identiques. le bloc est placé à gauche du bloc (ou du nœud) précédent.

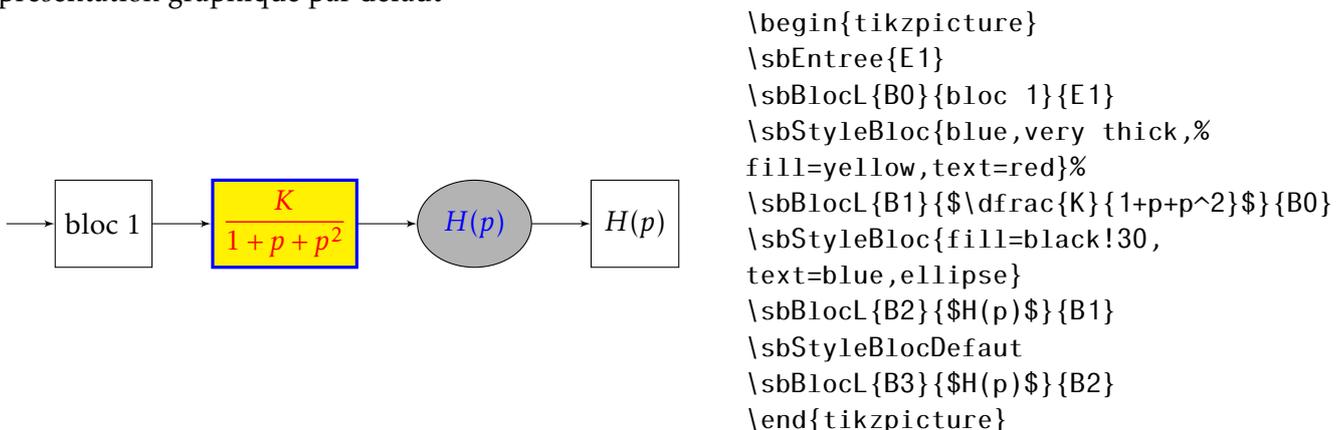
Bloc seul La commande `\sbBlocseul...` permet de tracer un bloc seul avec une entrée et une sortie cette commande est général utilisée seule. Le paramètre de distance est appliqué sur le lien d'entrée et de sortie

```
\sbBlocseul[3]{Entrée}{ $H(p)$ }{Sortie}
```

apporté

1.4.4 Personnalisation des blocs

La commande `\sbStyleBloc{liste d'option}` permet de modifier la représentation graphique du bloc (fond, couleur des traits, couleur du texte, épaisseur,...), les modifications sont valables jusqu'à une nouvelle définition. Les modifications de style apportée sont cumulatives. La commande `\sbStyleBlocDefault` ré-activer la représentation graphique par défaut



Comme on le voit ci-dessus, il même possible de modifier la forme du nœud. Toutes les options relatives au tracé des nœuds dans pgf/tikz sont utilisables.

1.5 Comparateur - Sommateur

1.5.1 Les commandes de base

```
\sbComp[distance]{nom}{bloc precedent}
\sbComp*[distance]{nom}{bloc precedent}
```

permettent de dessiner un comparateur, soit avec sa représentation usuelle  ou avec les opérateurs à

l'extérieur  avec les paramètres suivant :

étoile * : représentation usuelle sans l'étoile;

distance : paramètre optionnel permettant de positionner le comparateur par rapport au bloc précédent (la valeur par défaut est 4em);

nom : nom du comparateur, ce nom doit être unique dans votre schéma, il sert à référencer le comparateur pour tous les liens;

bloc precedent : le nom du bloc précédent, le comparateur est placé à sa droite, à la distance **distance**.

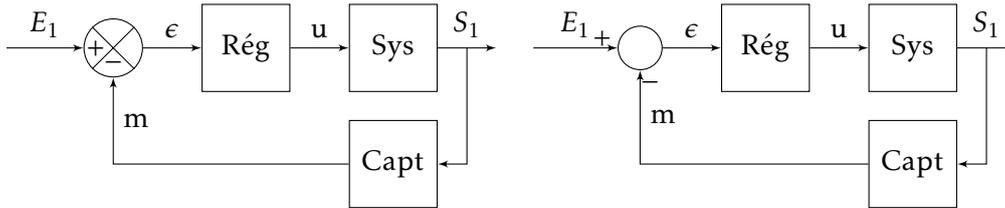
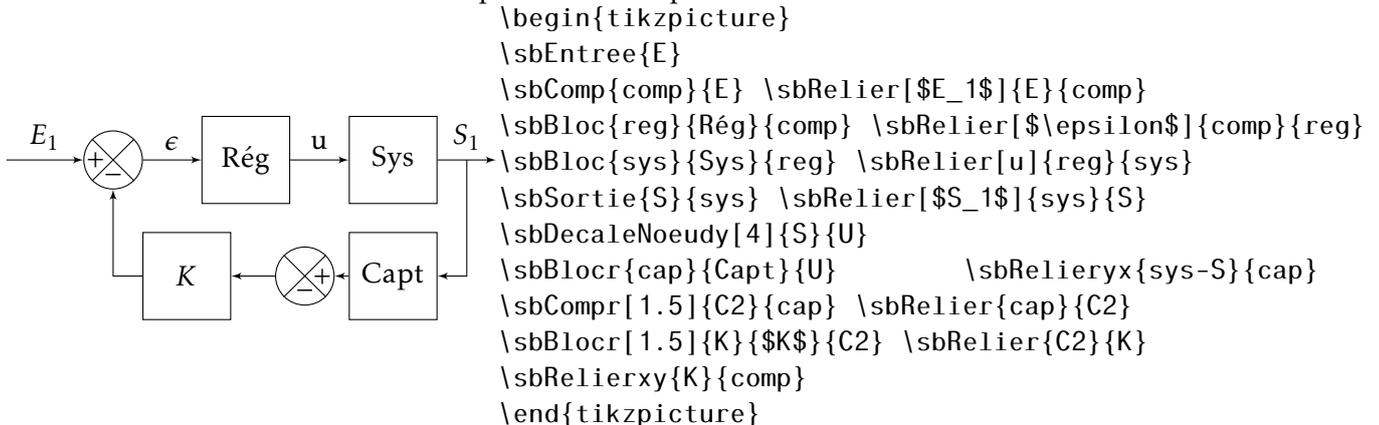


FIGURE 3 – Comparateur- commande étoilée ou non

1.5.2 Comparateur dans la boucle de retour

Les commandes indicées avec un « r » permettent de placer un correcteur dans la boucle de retour.



On peut ainsi installer toutes les formes de comparateurs et sommateurs.

1.5.3 Commandes et symboles

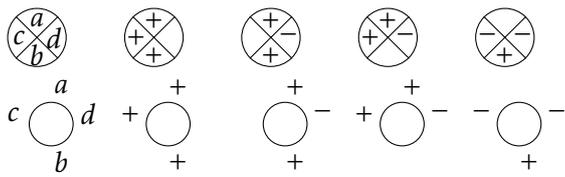
D'autres commandes génériques permettent de dessiner rapidement les principaux symboles de sommation et de comparaison.

	Comparateurs		Sommateurs	
Symbole				
code	<code>\sbComp{}</code> ...	<code>\sbComph{}</code> ...	<code>\sbSumb{}</code> ...	<code>\sbSumh{}</code> ...
Symbole				
code	<code>\sbComp*{}</code> ...	<code>\sbComph*{}</code> ...	<code>\sbSumb*{}</code> ...	<code>\sbSumh*{}</code> ...

TABLE 1 – Symboles et commandes de comparateurs et sommateurs

deux commandes plus génériques `\sbCompSum[dist]{nom}{E1}{a}{b}{c}{d}` et `\sbCompSumr[dist]{nom}{E1}{a}` permettent de dessiner tout sommateur et entre autres les sommateurs et comparateurs placés dans les lignes de

retour. les commandes étoilées `\sbCompSum* [dist] { nom } { E1 } { a } { b } { c } { d }` et `\sbCompSumr* [dist] { nom } { E1 } { a } { b }` permettent de représenter le sommateur (comparateur) avec les signes à l'extérieur.



```
\begin{tikzpicture}
\sbEntree{E1}
\sbCompSum[-4]{C1}{E1}{a}{b}{c}{d}
\sbCompSum[0]{C1}{E1}{+}{+}{+}{ }
\sbCompSum[4]{C1}{E1}{+}{+}{ }{-}
\sbCompSum[8]{C1}{E1}{+}{ }{+}{-}
\sbCompSum[12]{C1}{E1}{ }{+}{-}{-}
\end{tikzpicture}
```

1.6 Liens et renvois

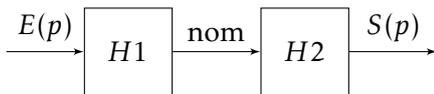
La commande `\sbBlocL . . .` permet de tracer un lien sans texte entre deux blocs successifs, les macros proposent trois autres types de liens qui permettent de tracer tout type de schéma.

- lien direct `\sbRelier [nom] { b1 } { b2 } ;`
- les renvois et sauts `\sbRenvoi { b1 } { b2 } { } ;`
- les liens décalés.

1.6.1 liens directs

Liens entre deux blocs (ou comparateur, entrée ou sortie) dans la chaîne directe (de la gauche vers la droite) ou dans la chaîne de retour (de la droite vers la gauche). Un lien direct entre deux blocs "b1" et "b2" s'écrit : `\sbRelier [nom] { b1 } { b2 }`

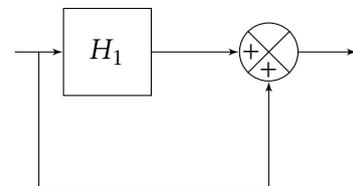
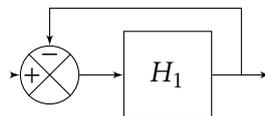
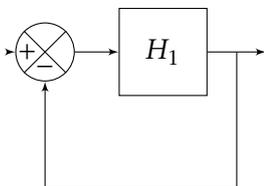
Le nom du lien (optionnel) est placé au dessus du lien au centre, Un nœud est associé au lien dans la figure sous le nom "b1-b2" (concaténation des noms avec un tiret "-")



```
\begin{tikzpicture}
\sbEntree{E1}
\sbBloc[3]{Bloc1}{H1}{E1}
\sbRelier[ $E(p)$ ]{E1}{Bloc1}
\sbBloc[3]{Bloc2}{H2}{Bloc1}
\sbRelier[nom]{Bloc1}{Bloc2}
\sbSortie[3]{S1}{Bloc2}
\sbRelier[ $S(p)$ ]{Bloc2}{S1}
\end{tikzpicture}
\end{tikzpicture}
```

1.6.2 Renvois et sauts

Les renvois sont des liens qui permettent soit de retourner en arrière soit de sauter un ou plusieurs blocs. Ils sont tracés entre un lien et un comparateur (sommateur);



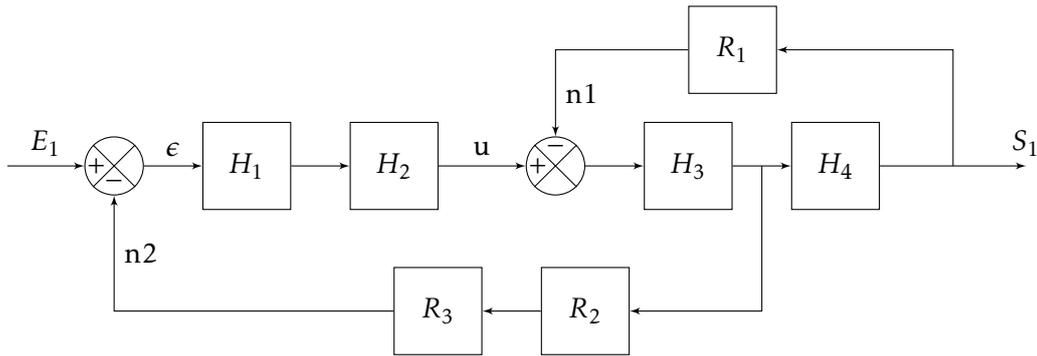
```
\begin{tikzpicture}
\sbEntree{E}
\sbComp[ 1.5 ]{comp}{E}
\sbRelier{E}{comp}
\sbBloc[ 1.5 ]{B}{H_1$}
{comp}
\sbRelier{comp}{B}
\sbSortie{S}{B}
\sbRelier{B}{S}
\sbRenvoi{B-S}{comp}{ }
\end{tikzpicture}
```

```
\begin{tikzpicture}
\sbEntree{E}
\sbComp[ 1.5 ]{comp}{E}
\sbRelier{E}{comp}
\sbBloc[ 1.5 ]{B}{H_1$}
{comp}
\sbRelier{comp}{B}
\sbSortie{S}{B}
\sbRelier{B}{S}
\sbRenvoi[-3]{B-S}{comp}{ }
\end{tikzpicture}
```

```
\begin{tikzpicture}
\sbEntree{E}
\sbBloc{B}{H_1$}{E}
\sbRelier{E}{B}
\sbSumb{sum}{B}
\sbRelier{B}{sum}
\sbSortie{S}{sum}
\sbRelier{sum}{S}
\sbRenvoi{E-B}{sum}{ }
\end{tikzpicture}
```

1.6.3 Liens décalés

Ces liens sont utilisés pour relier un bloc d'une ligne vers un bloc d'une ligne parallèle (vers l'avant ou l'arrière);



Code

```

\begin{tikzpicture}
\sbEntree{E}
\sbComp{a}{E}
\sbRelier[ $E_1$ ]{E}{a}
\sbBloc{b}{ $H_1$ }{a}
\sbRelier[ $\epsilon$ ]{a}{b}
\sbBlocL{c}{ $H_2$ }{b}
\sbComp{d}{c}
\sbRelier[u]{c}{d}
\sbBloc{e}{ $H_3$ }{d}
\sbBloc{f}{ $H_4$ }{e}
\sbSortie[5]{S1}{f}
\sbRelier{f}{S1}
\sbNomLien[0.8]{S1}{ $S_1$ }
\sbDecaleNoeudy[-4]{f}{u}
\sbDecaleNoeudy{e}{v}
\sbBlocr{r1}{ $R_1$ }{u}
\sbBlocr{r2}{ $R_2$ }{v}
\sbBlocrL{r3}{ $R_3$ }{r2}
\sbRelieryx{f-S1}{r1}
\sbRelieryx[n1]{r1}{d}
\sbRelieryx{e-f}{r2}
\sbRelieryx[n2]{r3}{a}
\end{tikzpicture}

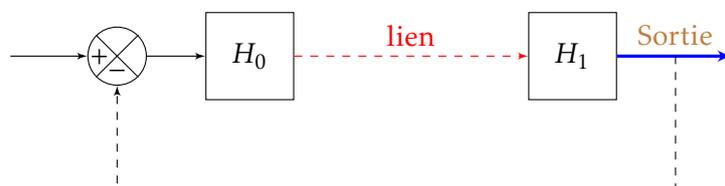
```

Commentaires

- Les premières commandes (de `\sbEntree...` à `\sbSortie{S1}...`) placent les blocs de la chaîne directe, certains blocs sont positionnés avec la commande `\sbBloc` d'autres avec `\sbBlocL`;
- le nom de la sortie n'est pas positionné avec la commande `\sbRelier{f}{S1}` mais avec la commande `\sbNomLien[0.8]{S1}{ S_1 }` afin qu'il ne soit pas superposé au trait du retour vers R_1 ;
- Les deux commandes `\sbDecaleNoeudy[-4]{f}{u}` et `\sbDecaleNoeudy{e}{v}` permettent de positionner le départ des deux boucles de retour, la première est décalé de [-4] au dessus de la chaîne précédente et positionnée par rapport au bloc `f`, la seconde est placée par défaut au dessous par rapport au bloc `e`. Ces nouveaux nœuds sont nommés `u` et `v`.
- Les blocs de retours sont ensuite tracés de la droite vers la gauche par rapport à ces nœuds avec la commande `\sbBlocr...` et `\sbBlocrL...` puis reliés.
- La commande `\sbRelieryx{f-S1}{r1}` trace le lien depuis le milieu de la liaison entre les blocs `f` et `s` et le bloc `r1`, la commande `\sbRelieryx[n1]{r1}{d}` trace le lien vers le comparateur

1.6.4 Personnalisation des liens

La commande `\sbStyleLien{liste d'option}` permet de personnaliser le tracé des liens et des textes associés.



```

\begin{tikzpicture}
\sbEntree{E}
\sbComp{comp}{E}
\sbRelier{E}{comp}
\sbBlocL{B1}{\$H_0\$}{comp}
\sbStyleLien{dashed, red}
\sbBloc[8]{B2}{\$H_1\$}{B1}
\sbRelier[lien]{B1}{B2}

\sbSortie[4]{S}{B2}
\sbStyleLienDefault
\sbStyleLien{blue, very thick, text=brown}
\sbRelier[Sortie]{B2}{S}
\sbStyleLien{dashed}
\sbRenvoi{B2-S}{comp}{}
\sbStyleLienDefault
\end{tikzpicture}

```

Cette commande s'applique à tous les types de liens, les styles sont actifs jusqu'à une nouvelle définition, et comme pour la commande `\sbStyleBloc{liste d'option}` ils sont cumulatifs. La commande `\sbStyleLienDefault` ré-initialise le style par défaut.

1.7 Chaînes et Boucles

Plusieurs commandes globales facilitent le tracé de schéma blocs, elle tracent directement des chaînes de blocs, ou des boucles.

1.7.1 Chaînes

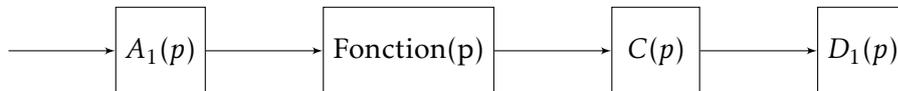
deux commandes, une qui pour trace des chaînes directe (de la gauche vers la droite), l'autre les chaînes de retour (de la droite vers la gauche).

Chaîne directe : `\sbChaine[pas]{Noeud depart}{liste blocs Nom/Fonction}`

```

\begin{tikzpicture}
\sbEntree{E}
\sbChaine[4]{E}{A/\$A_1(p)\$,B/Fonction(p),C/\$C(p)\$,D/\$D_1(p)\$}
\end{tikzpicture}

```

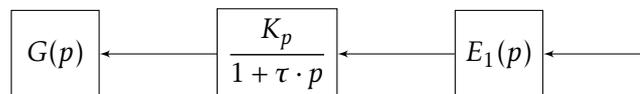


Chaîne de retour : `\sbChaineRetour[pas]{Noeud depart}{liste blocs Nom/Fonction}`

```

\begin{tikzpicture}
\sbEntree{E}
\sbChaineRetour[3]{E}{E1/\$E_1(p)\$,F/\$\dfrac{K_p}{1+\tau \cdot p}\$,G/\$G(p)\$}
\end{tikzpicture}

```



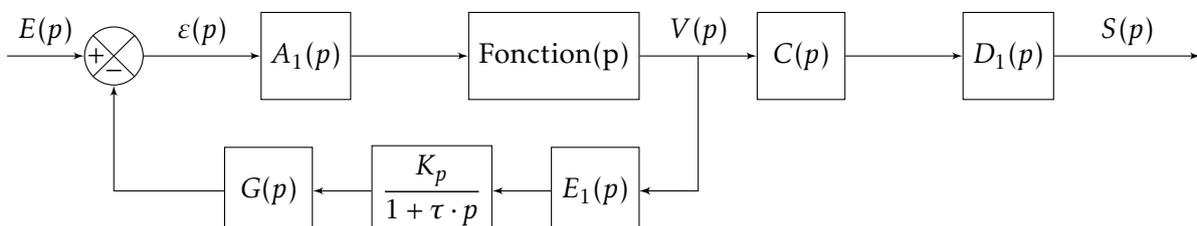
On retrouve dans ces deux commandes

pas : distance en em entre les blocs,

noeud depart : le premier bloc de la chaîne est positionnée à la distance pas du noeud noeud depart,

liste blocs : la liste des blocs est notée avec la syntaxe suivante $Nom_1/Fonction_1, Nom_2/Fonction_2, \dots, Nom_i/Fonction_i, \dots, Nom_n/Fonction_n$.

Avec ces deux commandes, on dessine rapidement le schéma ci dessous avec un code relativement court.



```

\begin{tikzpicture}
\sbEntree{E}
\sbComp{comp}{E}\sbRelier[ $E(p)$ ]{E}{comp}
\sbChaine[4]{comp}{A/ $A_1(p)$ ,B/Fonction(p),C/ $C(p)$ ,D/ $D_1(p)$ }
\sbDecaleNoeudy[5]{B-C}{DebutRetour}
\node[above of=comp,node distance=0.5em]{ $\varepsilon(p)$ };
\node[above of=B-C,node distance=0.5em]{ $V(p)$ };
\sbChaineRetour[2]{DebutRetour}{E1/ $E_1(p)$ ,F/ $\frac{K_p}{1+\tau \cdot p}$ ,G/ $G(p)$ }
\sbRelieryx{B-C}{E1}
\sbRelieryx{G}{comp}
\sbSortie[5]{S}{D}\sbRelier[ $S(p)$ ]{D}{S}
\end{tikzpicture}

```

On remarquera, tous les liens sont nommés, le nom est construit à partir du nom des deux blocs à relier, ainsi, B-C correspond au point milieu du lien entre B et C.

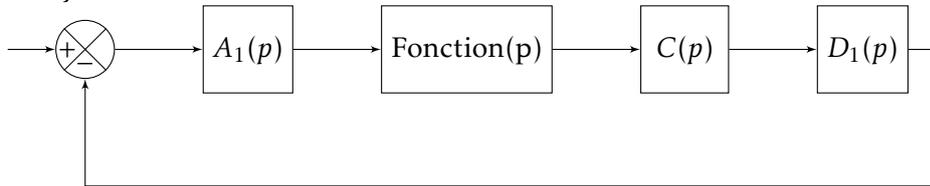
La commande tikz `\node[above of=B-C,node distance=0.5em]{ $V(p)$ };` permet de placer le nom du lien au dessus (à la distance 0.5em) de celui-ci.

1.7.2 Boucles

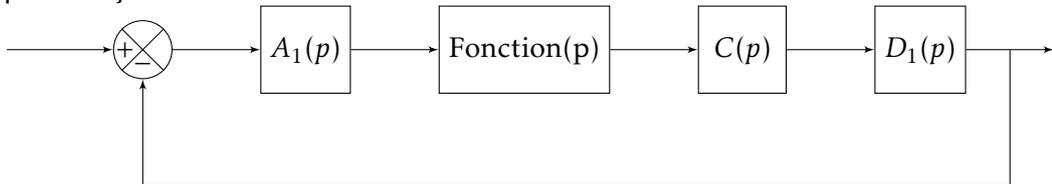
Trois commandes permettent de dessiner des schémas blocs bouclés unitaire ou non.

Boucle unitaire : deux commandes, une pour dessiner une boucle à retour unitaire à insérer dans un schéma, l'autre pour dessiner complètement la boucle.

— `\sbBoucle[pas]{noeud depart}{liste de blocs chaine directe}` ou
`\begin{tikzpicture}`
`\sbEntree{E}`
`\sbBoucle[3]{E}{A/ $A_1(p)$,B/Fonction(p),C/ $C(p)$,D/ $D_1(p)$ }`
`\end{tikzpicture}`



— `\sbBoucleSeule[pas]{Noeud debut}{liste blocs}{Sortie}`
`\begin{tikzpicture}`
`\sbEntree{E}`
`\sbBoucleSeule[3]{E}{A/ $A_1(p)$,B/Fonction(p),C/ $C(p)$,D/ $D_1(p)$ }`
`\end{tikzpicture}`

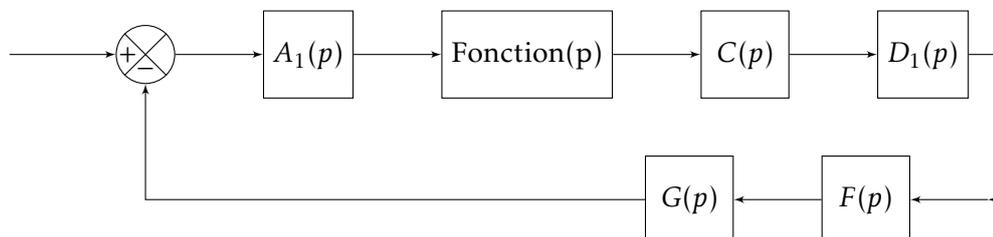


Boucle de retour non unitaire : `\sbBoucleRetour[pas]{noeud depart}{liste chaine directe}{liste chain`

```

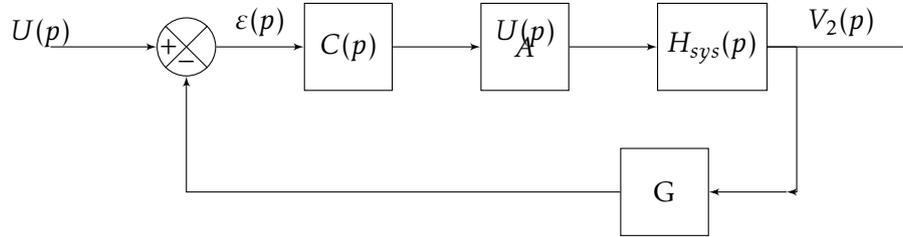
\begin{tikzpicture}
\sbEntree{E}
\sbBoucleRetour[3]{E}{A/ $A_1(p)$ ,B/Fonction(p),C/ $C(p)$ ,D/ $D_1(p)$ }{F/ $F(p)$ ,G/ $G(p)$ }
\end{tikzpicture}

```



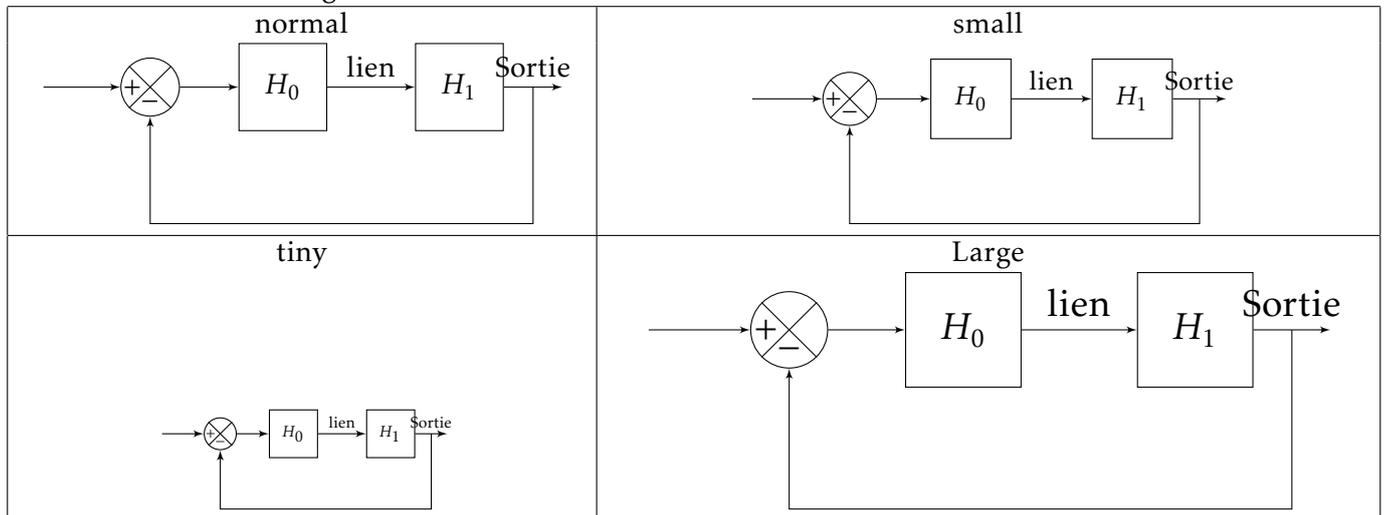
Encore un petit exemple, dans le quel on peut noter que le lien entre le comparateur et le bloc suivant est nommé CompU-C, ce nom est construit en concaténant le mot *Comp* avec le nom du nœud de départ *U*, le tiret - et le nom du nœud suivant *C*.

```
\begin{tikzpicture}
\sbEntree{U}
\sbBoucleRetour[3]{U}{C/$C(p)$,A/$A$,H/$H_{sys}(p)$}{G/G}
\node[above of=CompU-C,node distance=0.5em]{$\varepsilon(p)$};
\node[above of=A,node distance=0.5em]{$U(p)$};
\node[above of=U,node distance=0.5em]{$U(p)$};
\sbSortie[5]{V2}{H}\sbRelier[$V_2(p)$]{H}{V2}
\end{tikzpicture}
```

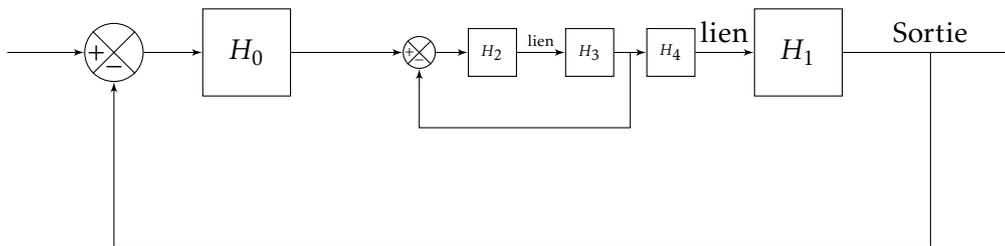


1.8 Changement de taille

Toutes dimensions étant en em, le changement de taille d'un graphe se fait en changeant la taille des caractères de l'environnement englobant.



Il est aussi possible d'utiliser ce changement de taille pour par exemple insister sur une partie de la boucle comme par exemple sur l'exemple ci-dessous, où on souhaite insister sur la boucle extérieure.

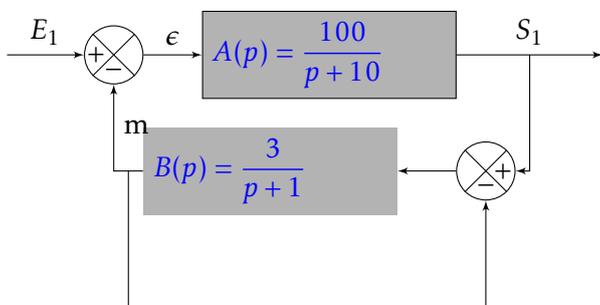


```
\begin{tikzpicture}
\sbEntree{E}
\sbComp{comp}{E}
\sbRelier{E}{comp}
\sbBlocL{B1}{$H_0$}{comp}
\begin{tiny}
\sbComp[8]{comp2}{B1}
\sbRelier{B1}{comp2}
\sbBlocL{B10}{$H_2$}{comp2}
\sbBloc[3]{B20}{$H_3$}{B10}
\sbRelier[lien]{B10}{B20}
\sbBloc{B30}{$H_4$}{B20}
\sbRelier{B20}{B30}
\sbRenvoi{B20-B30}{comp2}{}
\end{tiny}
\end{tikzpicture}
```

2 Installation

- Copier le package schemabloc.sty dans votre répertoire localtexmf/tex/latex ...
 - placer dans l'entête de votre document `\usepackage{schemabloc}`
- Vous pouvez aussi copier le code des macros ci-dessous dans votre entête.

3 Exemples



```

\begin{tikzpicture}
\sbEntree{E}
\sbStyleBloc{fill=black!30,text=blue,text width=8em}
\sbComp{comp}{E} \sbRelier[{$E_1$}]{E}{comp}
\sbBloc{A}{A(p)=\dfrac{100}{p+10}}{comp} \sbRelier[{\epsilon}]{comp}{A}
\sbSortie[5]{S}{A} \sbRelier[{$S_1$}]{A}{S}
\sbDecaleNoeudy[4]{S}{U}
\sbCompSum[-4]{C1}{U}{-}{+} \sbRelieryx{A-S}{C1}
\sbBlocr{B}{B(p)=\dfrac{3}{p+1}}{C1} \sbRelier{C1}{B}

\sbRelieryx[m]{B}{comp}
\sbRenvoi{B-comp-1}{C1}{}
\end{tikzpicture}

```

4 Code des macros