

# Une application des S-langages en musique et leur perspective

Antoine Allombert

MaMux

Ircam

12 mars 2010

# Plan

- 1 Probématique
- 2 Vers un formalisme de partitions interactives
- 3 Réseaux d'occurrences et S-langages
- 4 Démonstration
- 5 Perspectives

<http://recherche.ircam.fr/equipes/repmus/allombert/interactive-scores.html>

# Problématique

- Développer un système de partitions pour la composition et l'interprétation.
- Deux phases distinctes :
  - La composition :
  - L'interprétation :

# Problématique

- Développer un système de partitions pour la composition et l'interprétation.
- Deux phases distinctes :
  - La composition :
    - créer un environnement de composition
  - L'interprétation :

# Problématique

- Développer un système de partitions pour la composition et l'interprétation.
- Deux phases distinctes :
  - La composition :
    - créer un environnement de composition
    - utiliser des contraintes temporelles pour définir la cohérence globale de la pièce.
  - L'interprétation :

# Problématique

- Développer un système de partitions pour la composition et l'interprétation.
- Deux phases distinctes :
  - La composition :
    - créer un environnement de composition
    - utiliser des contraintes temporelles pour définir la cohérence globale de la pièce.
  - L'interprétation :
    - permettre au musicien de profiter des libertés octroyées par le compositeur

# Problématique

- Développer un système de partitions pour la composition et l'interprétation.
- Deux phases distinctes :
  - La composition :
    - créer un environnement de composition
    - utiliser des contraintes temporelles pour définir la cohérence globale de la pièce.
  - L'interprétation :
    - permettre au musicien de profiter des libertés octroyées par le compositeur
    - maintenir les limites imposées par le compositeur

# Problématique

- Développer un système de partitions pour la composition et l'interprétation.
- Deux phases distinctes :
  - La composition :
    - créer un environnement de composition
    - utiliser des contraintes temporelles pour définir la cohérence globale de la pièce.
  - L'interprétation :
    - permettre au musicien de profiter des libertés octroyées par le compositeur
    - maintenir les limites imposées par le compositeur
    - limiter des interactions aux déclenchements et relâchements (contrôles discrets)

# Problématique

- Développer un système de partitions pour la composition et l'interprétation.
- Deux phases distinctes :
  - La composition :
    - créer un environnement de composition
    - utiliser des contraintes temporelles pour définir la cohérence globale de la pièce.
  - L'interprétation :
    - permettre au musicien de profiter des libertés octroyées par le compositeur
    - maintenir les limites imposées par le compositeur
    - limiter des interactions aux déclenchements et relâchements (contrôles discrets)
- un système générique

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# L'interprétation

- Formalisation (Jean Haury [Hau87])
  - l'articulation
  - l'accentuation
  - les variations dynamiques
  - les modifications agogiques
- Points d'interaction
- Ecriture des libertés et des limites
  - implicite (ordre des notes)
  - explicite (indication de tempo, point d'orgue)

# Modifications agogiques

- contraintes qualitatives
- contraintes quantitatives

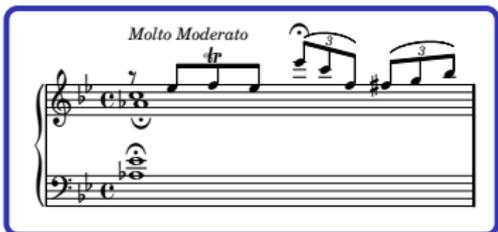


Figure: Un extrait de la *Rhapsody in Blue* de Georges Gershwin

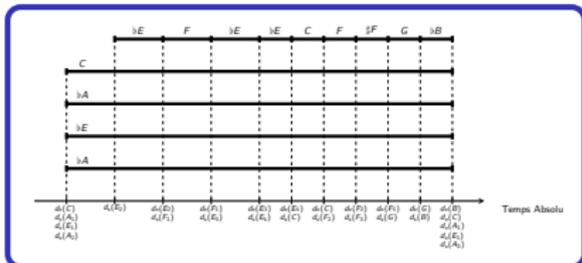


Figure: Diagramme temporel sans interprétation

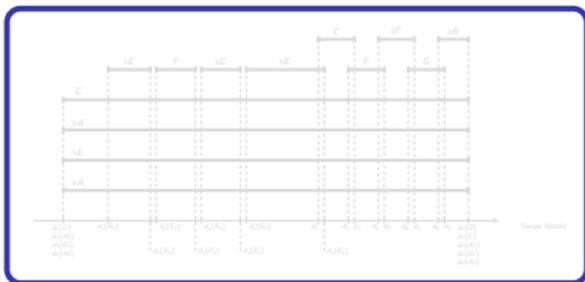


Figure: Diagramme temporel avec *staccato* et *legato*

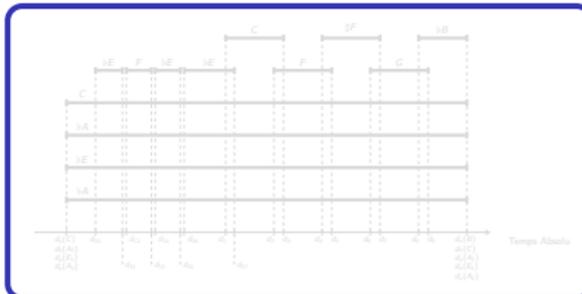


Figure: Diagramme temporel avec changement de tempo

# Modifications agogiques

- contraintes qualitatives
- contraintes quantitatives

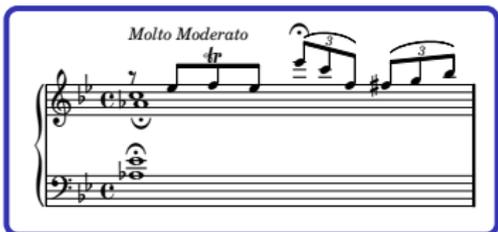


Figure: Un extrait de la *Rhapsody in Blue* de Georges Gershwin

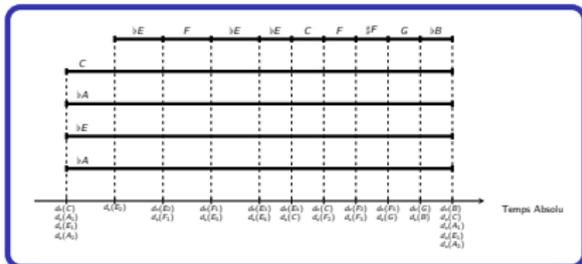


Figure: Diagramme temporel sans interprétation

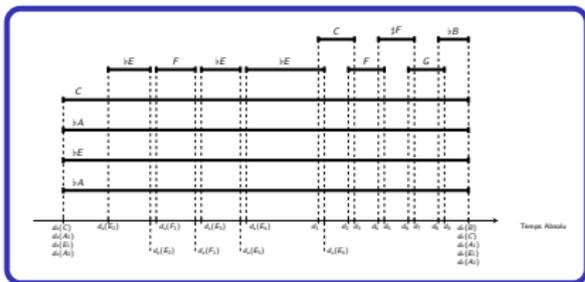


Figure: Diagramme temporel avec *staccato* et *legato*



Figure: Diagramme temporel avec changement de tempo

# Modifications agogiques

- contraintes qualitatives
- contraintes quantitatives

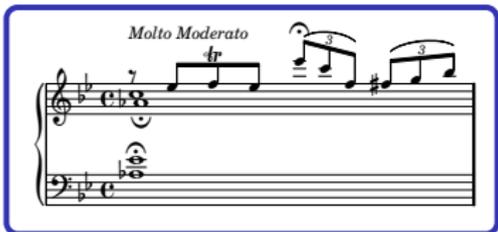


Figure: Un extrait de la *Rhapsody in Blue* de Georges Gershwin

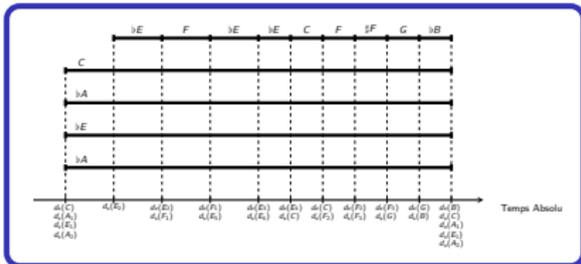


Figure: Diagramme temporel sans interprétation

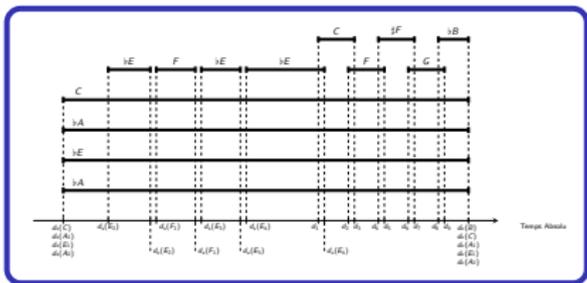


Figure: Diagramme temporel avec *staccato* et *legato*

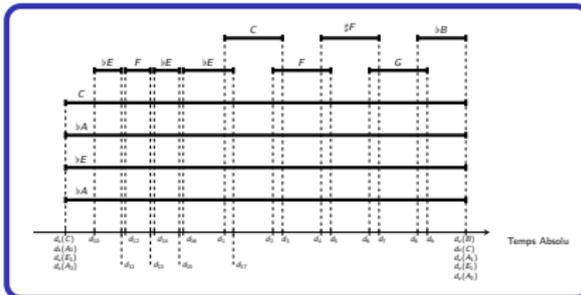
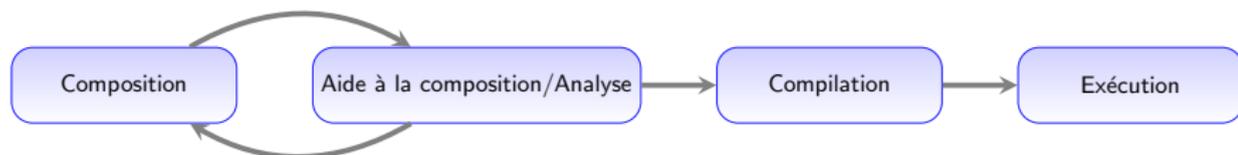


Figure: Diagramme temporel avec changement de tempo

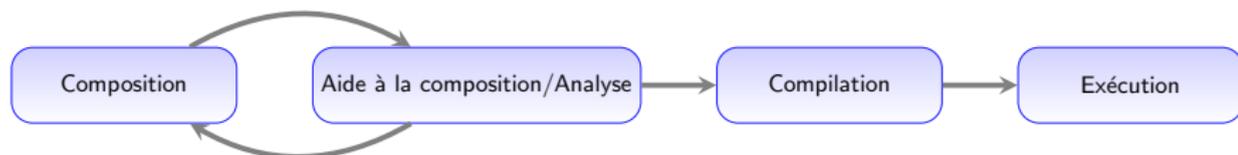
# Processus de composition



## • De quoi avons-nous besoin ?

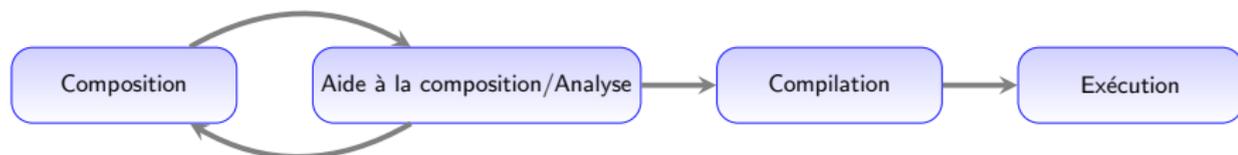
- Un formalisme d'écriture de partitions interactives utilisable par les compositeurs
- Une machine d'exécution des partitions
- Une méthode de compilation depuis le formalisme d'écriture vers le langage de la machine

# Processus de composition



- De quoi avons-nous besoin ?
  - Un formalisme d'écriture de partitions interactives utilisable par les compositeurs
  - Une machine d'exécution des partitions
  - Une méthode de compilation depuis le formalisme d'écriture vers le langage de la machine

# Processus de composition



- De quoi avons-nous besoin ?
  - Un formalisme d'écriture de partitions interactives utilisable par les compositeurs
  - Une machine d'exécution des partitions
  - Une méthode de compilation depuis le formalisme d'écriture vers le langage de la machine

# Processus de composition



- De quoi avons-nous besoin ?
  - Un formalisme d'écriture de partitions interactives utilisable par les compositeurs
  - Une machine d'exécution des partitions
  - Une méthode de compilation depuis le formalisme d'écriture vers le langage de la machine

# Objets temporels

- Représentation hiérarchique sous forme de boîtes
  - Maquettes d'*Open Music*
  - *Boxes*
- objet temporel :

$$OT = \langle t, r, p, E, S, \mathcal{P}, \mathcal{V}, \mathcal{E}, \mathcal{B}, \mathcal{R}, \mathcal{C} \rangle$$

- $p$  : un processus, opération réalisée par OT.
- $\mathcal{P}$  : des points de contrôle, événements distingués de l'exécution de OT.

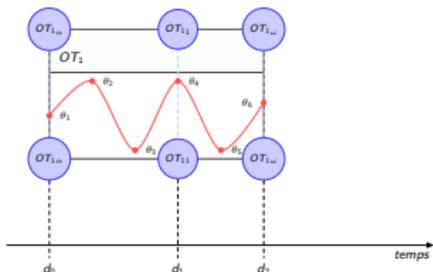


Figure: Un objet simple

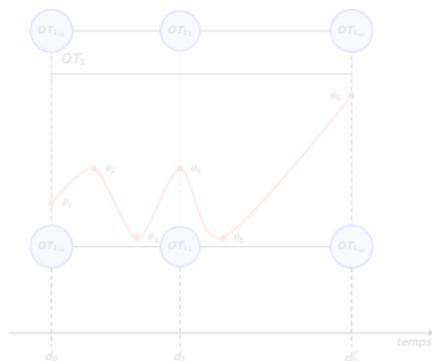


Figure: Une autre mise en temps

# Objets temporels

- Représentation hiérarchique sous forme de boîtes
  - Maquettes d'*Open Music*
  - *Boxes*
- objet temporel :

$$OT = \langle t, r, p, E, S, \mathcal{P}, \mathcal{V}, \mathcal{E}, \mathcal{B}, \mathcal{R}, \mathcal{C} \rangle$$

- $p$  : un processus, opération réalisée par OT.
- $\mathcal{P}$  : des points de contrôle, événements distingués de l'exécution de OT.

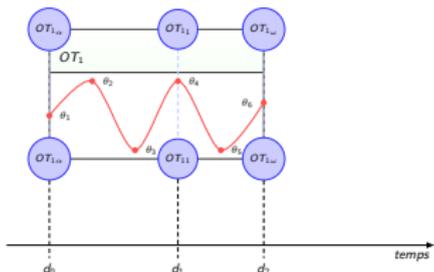


Figure: Un objet simple

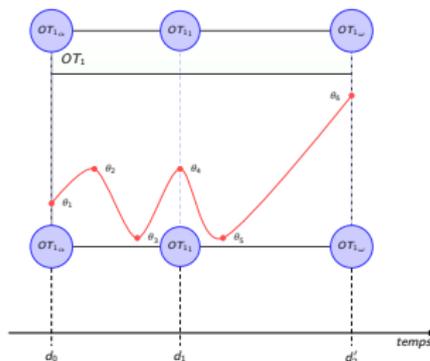


Figure: Une autre mise en temps

# Structures linéaires et relations temporelles

$$OT = \langle t, r, p, E, S, \mathcal{P}, \mathcal{V}, \mathcal{E}, \mathcal{B}, \mathcal{R}, \mathcal{C} \rangle$$

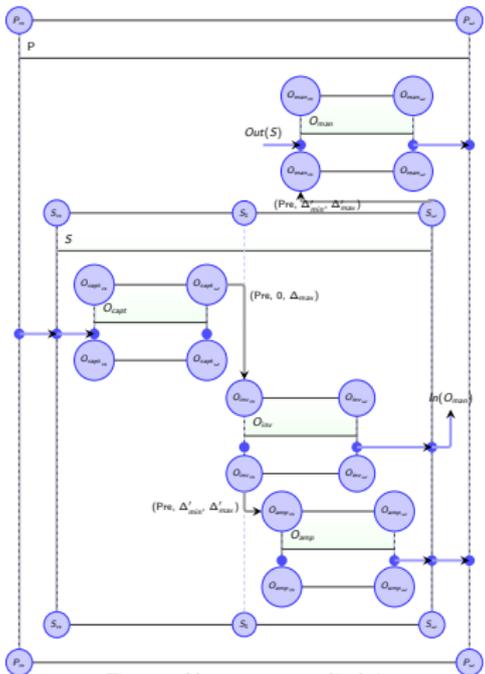


Figure: Une structure linéaire

- $\mathcal{R}$  : relations temporelles
  - **pre** : relation de précédence
  - **post** : relation de postériorité
  
- une relation définit un **intervalle**
  
- $r$  : rapport quanta temporels :  
 $q(S) = r(S).q(\text{parent}(S))$

# Structures linéaires et relations temporelles

$$OT = \langle t, r, p, E, S, \mathcal{P}, \mathcal{V}, \mathcal{E}, \mathcal{B}, \mathcal{R}, \mathcal{C} \rangle$$

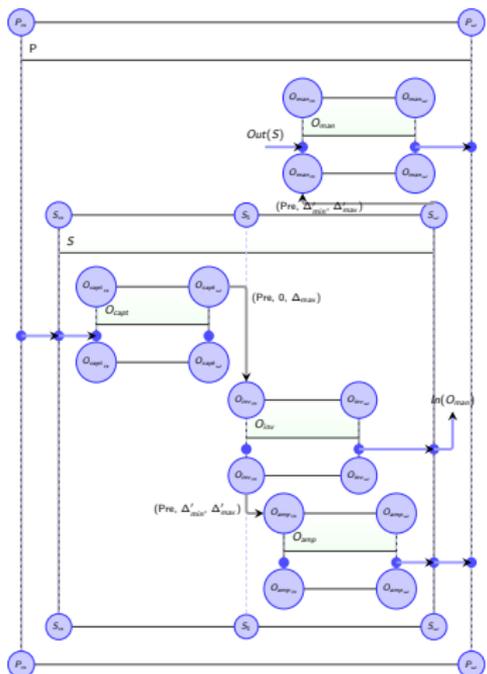


Figure: Une structure linéaire

- $\mathcal{R}$  : relations temporelles
  - **pre** : relation de précédence
  - **post** : relation de postériorité
  
- une relation définit un **intervalle**
  
- $r$  : rapport quanta temporels :  
 $q(S) = r(S).q(\text{parent}(S))$

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

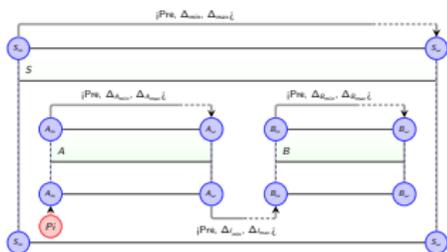


Figure: Point d'orgue

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

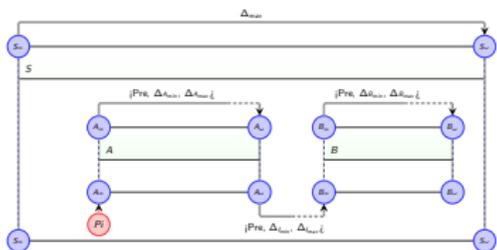


Figure: Point d'orgue

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

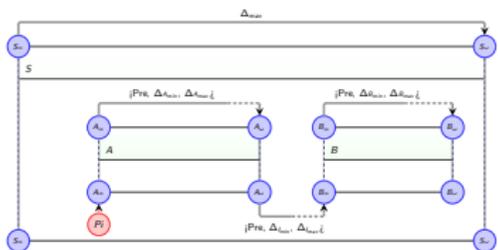


Figure: Point d'orgue

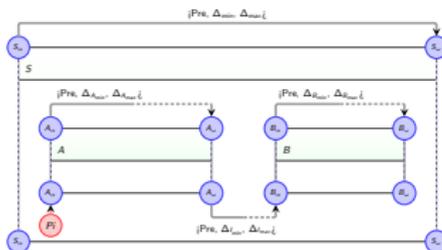


Figure: Ecrasement proportionnel

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

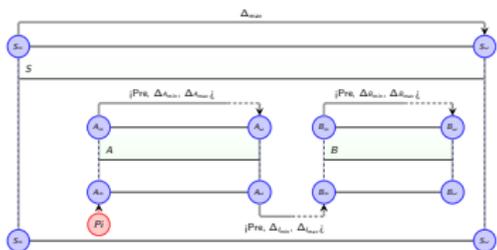


Figure: Point d'orgue

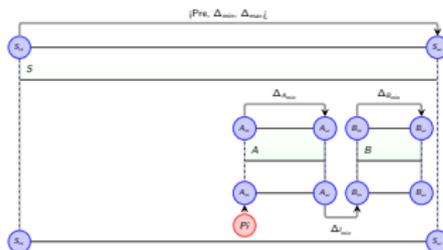


Figure: Ecrasement proportionnel

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

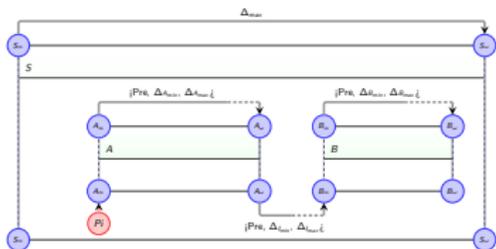


Figure: Point d'orgue

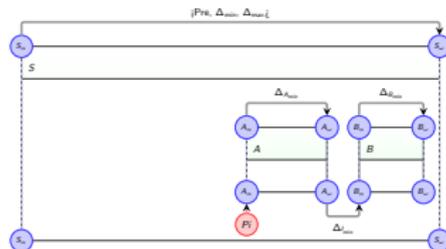


Figure: Ecrasement proportionnel

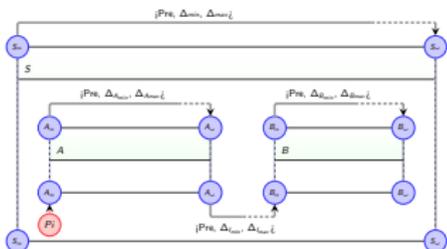


Figure: Ecrasement chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

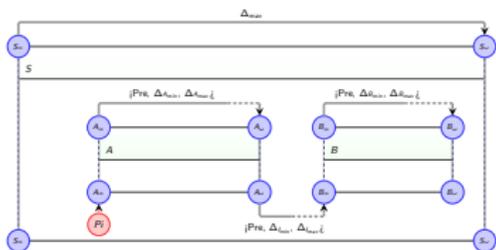


Figure: Point d'orgue

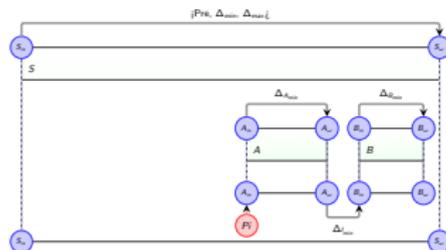


Figure: Ecrasement proportionnel

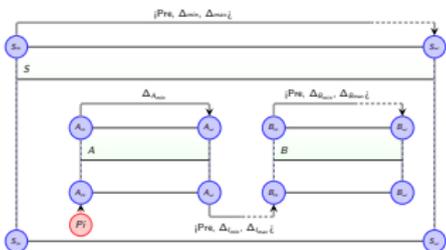


Figure: Ecrasement chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

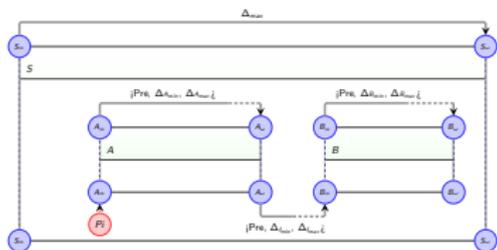


Figure: Point d'orgue

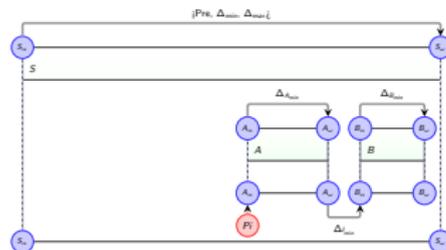


Figure: Ecrasement proportionnel

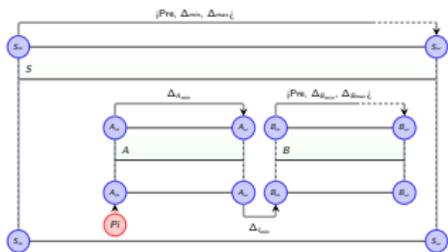


Figure: Ecrasement chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

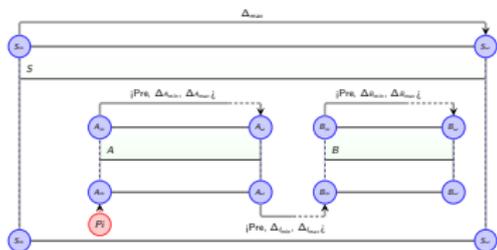


Figure: Point d'orgue

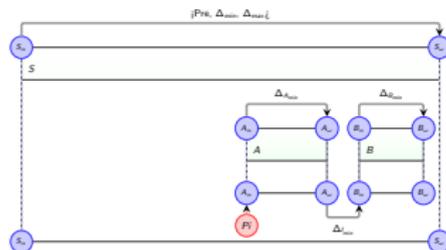


Figure: Ecrasement proportionnel

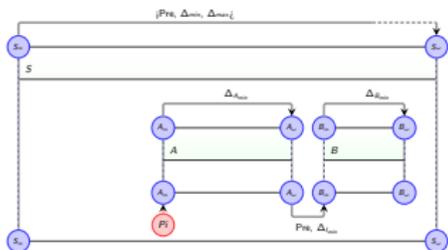


Figure: Ecrasement chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

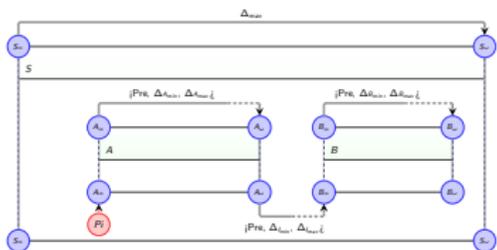


Figure: Point d'orgue

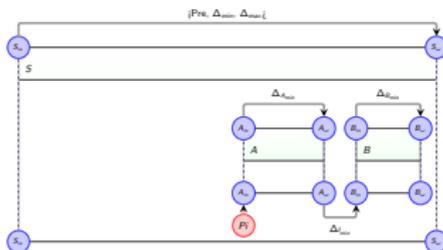


Figure: Ecrasement proportionnel

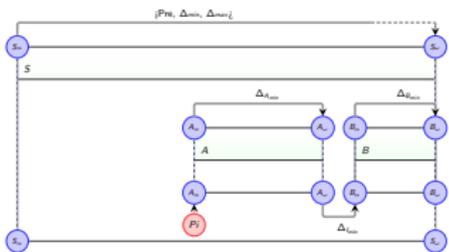


Figure: Ecrasement chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

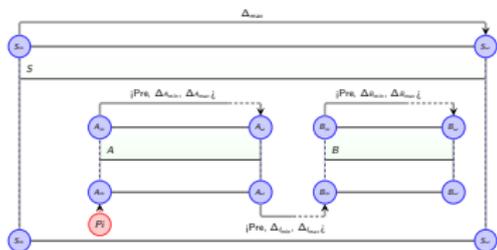


Figure: Point d'orgue

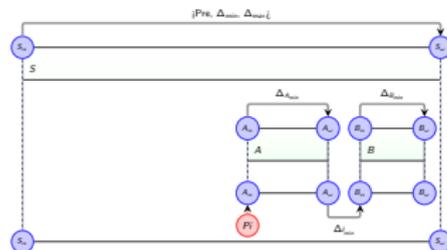


Figure: Ecrasement proportionnel

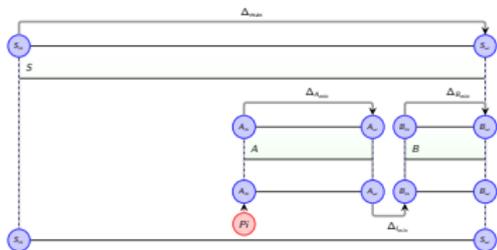


Figure: Ecrasement chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

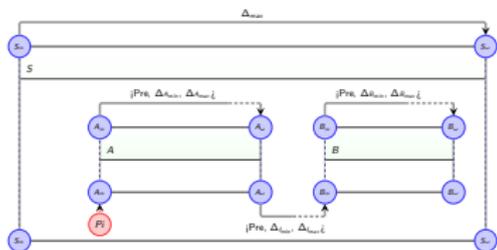


Figure: Point d'orgue

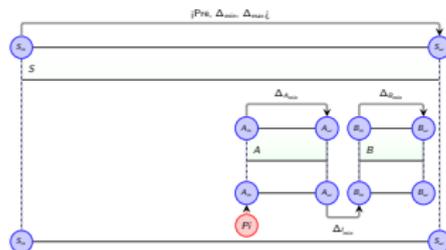


Figure: Ecrasement proportionnel

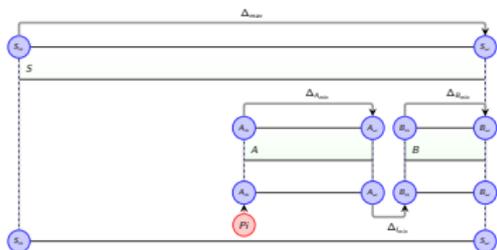


Figure: Ecrasement chronologique

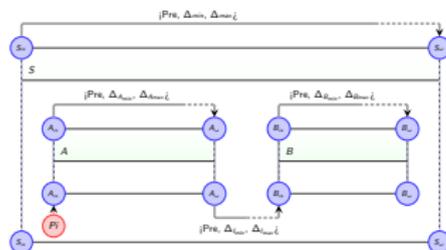


Figure: Ecrasement anti-chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

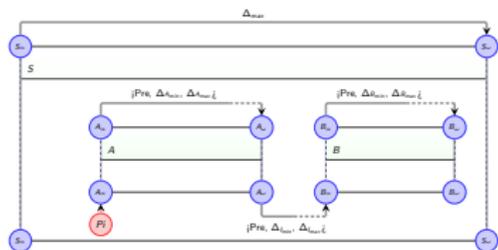


Figure: Point d'orgue

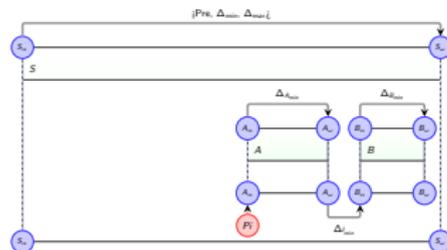


Figure: Ecrasement proportionnel

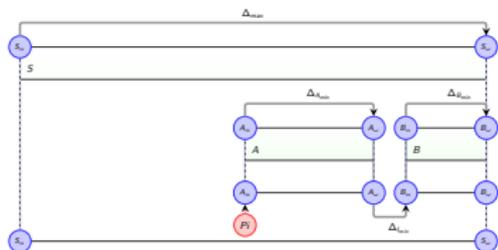


Figure: Ecrasement chronologique

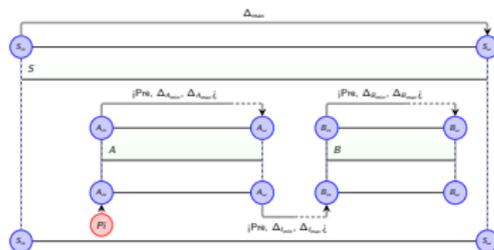


Figure: Ecrasement anti-chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

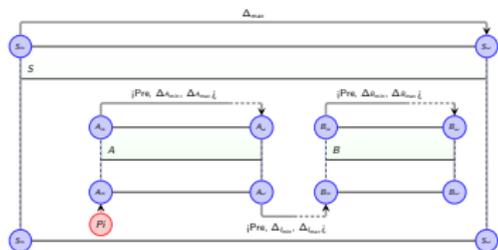


Figure: Point d'orgue

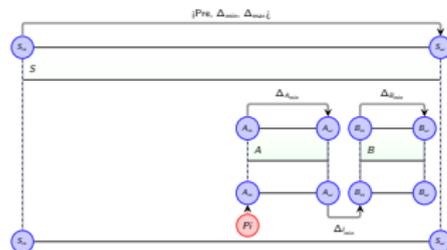


Figure: Ecrasement proportionnel

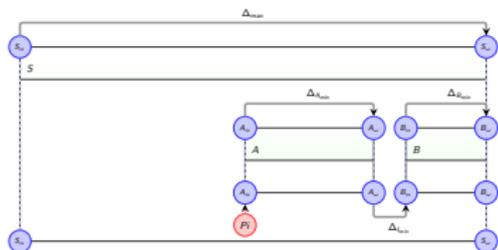


Figure: Ecrasement chronologique

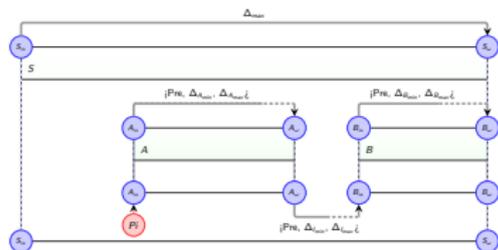


Figure: Ecrasement anti-chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

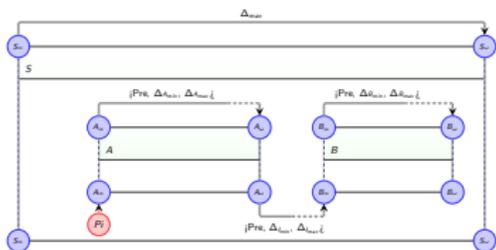


Figure: Point d'orgue

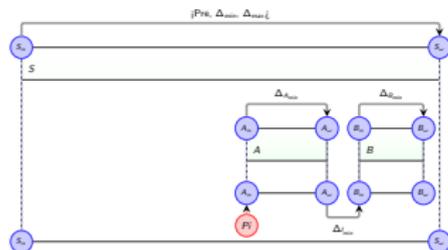


Figure: Ecrasement proportionnel

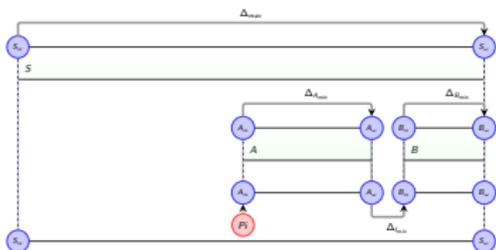


Figure: Ecrasement chronologique

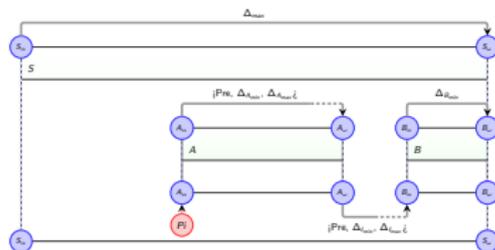


Figure: Ecrasement anti-chronologique

# Points d'interaction

- rendre dynamique le déclenchement de certains points de contrôle
- contraintes d'intervalles définissent des stratégies d'adaptation

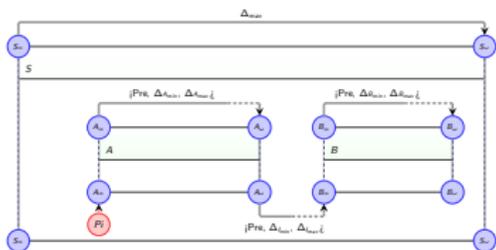


Figure: Point d'orgue

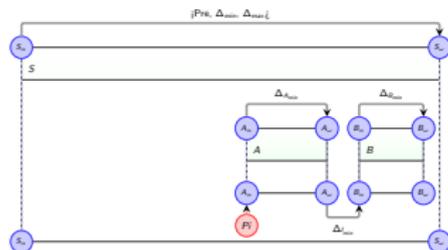


Figure: Ecrasement proportionnel

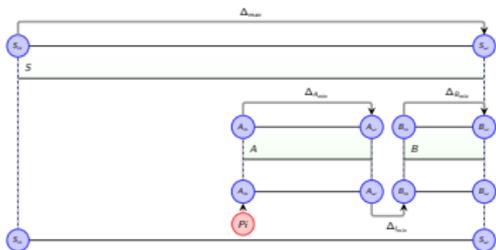


Figure: Ecrasement chronologique

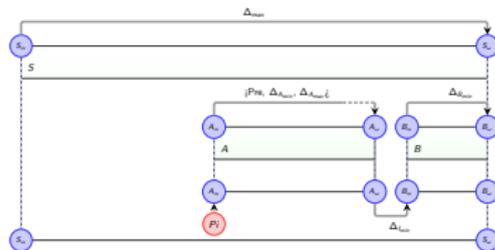


Figure: Ecrasement anti-chronologique

# La machine *ECO*

- *E (Environment)* : un environnement musical
- *C (Controls)* : un flux d'événements datés fournis en entrée
- *O (Outputs)* : un flux de sortie

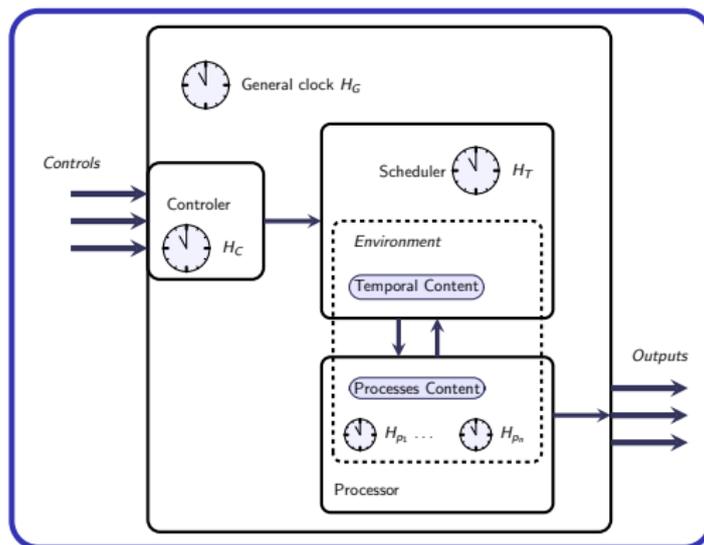


Figure: Architecture de la machine *ECO*

# Machine *ECO*

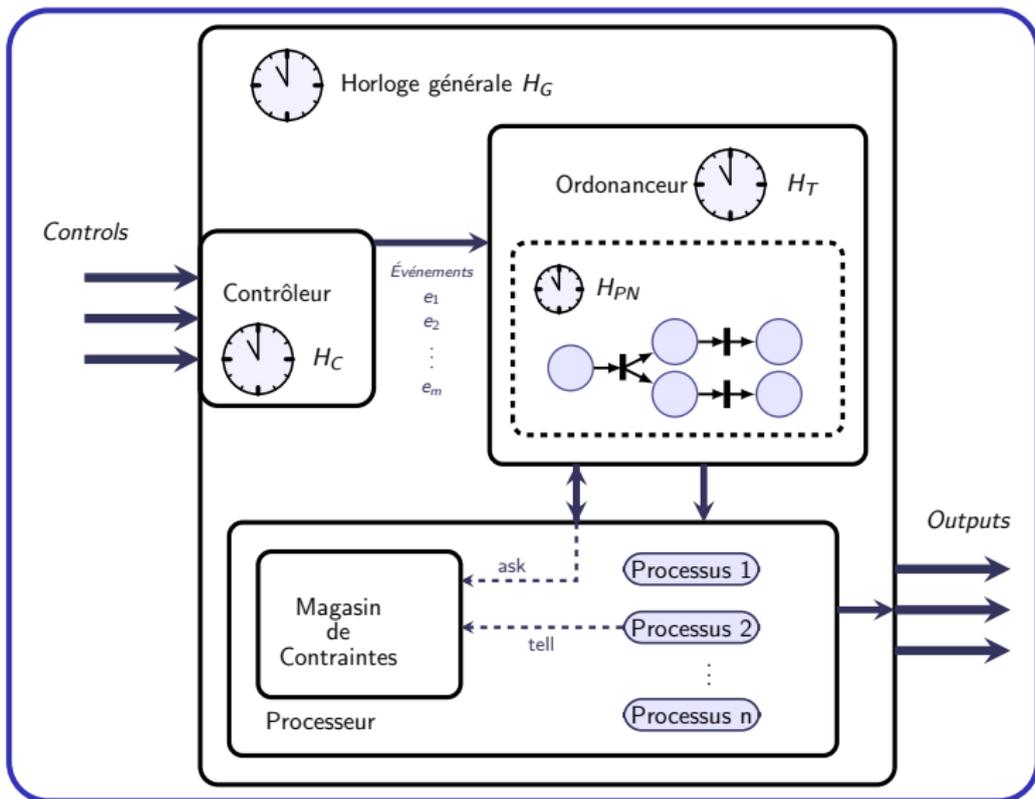


Figure: Architecture de la machine *ECO* pour le modèle du point d'orgue

# Réseaux de Petri et réseaux d'occurrences

## • Réseaux de Petri

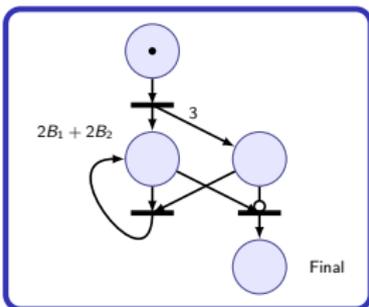
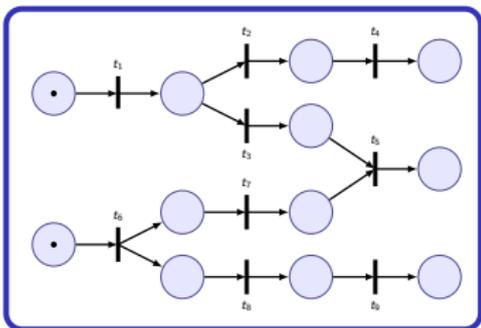


Figure: Modélisation du *Boléro* de Ravel en réseau de Petri [HR88]

## • Réseaux d'occurrences



- **précédence** :  $t_1 \leq t_2, t_8 \leq t \dots$
- **concurrence** :  $t_1 || t_6, t_7 || t_8 \dots$
- **conflit** :  $t_2 \# t_3, t_2 \# t_5, t_3 \# t_4$

# Réseaux de Petri et réseaux d'occurrences

## • Réseaux de Petri

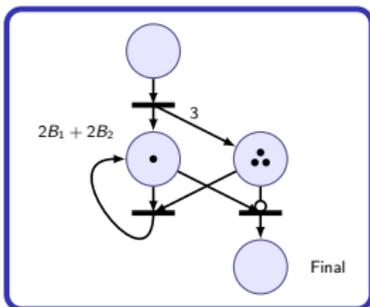
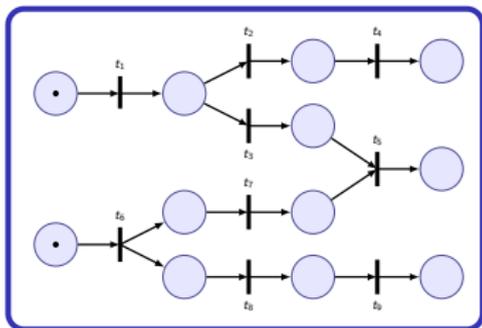


Figure: Modélisation du *Boléro* de Ravel en réseau de Petri [HR88]

## • Réseaux d'occurrences



- **précédence** :  $t_1 \leq t_2, t_8 \leq t \dots$
- **concurrence** :  $t_1 || t_6, t_7 || t_8 \dots$
- **conflit** :  $t_2 \# t_3, t_2 \# t_5, t_3 \# t_4$

# Réseaux de Petri et réseaux d'occurrences

## • Réseaux de Petri

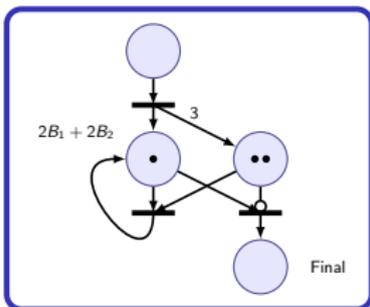
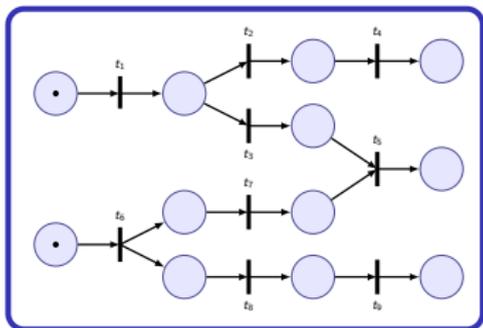


Figure: Modélisation du *Boléro* de Ravel en réseau de Petri [HR88]

## • Réseaux d'occurrences



- **précédence** :  $t_1 \leq t_2, t_8 \leq t \dots$
- **concurrence** :  $t_1 || t_6, t_7 || t_8 \dots$
- **conflit** :  $t_2 \# t_3, t_2 \# t_5, t_3 \# t_4$

# Réseaux de Petri et réseaux d'occurrences

## • Réseaux de Petri

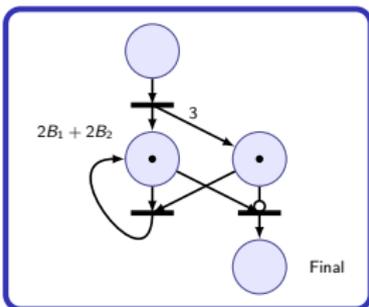
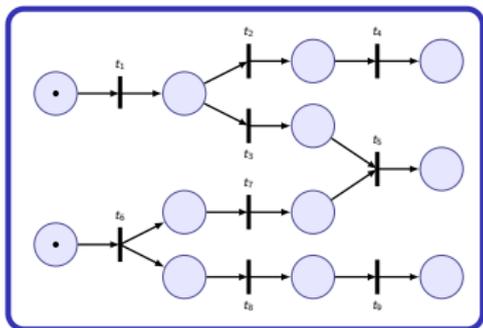


Figure: Modélisation du *Boléro* de Ravel en réseau de Petri [HR88]

## • Réseaux d'occurrences



- précedence :  $t_1 \leq t_2, t_8 \leq t \dots$
- concurrence :  $t_1 || t_6, t_7 || t_8 \dots$
- conflit :  $t_2 \# t_3, t_2 \# t_5, t_3 \# t_4$

# Réseaux de Petri et réseaux d'occurrences

## • Réseaux de Petri

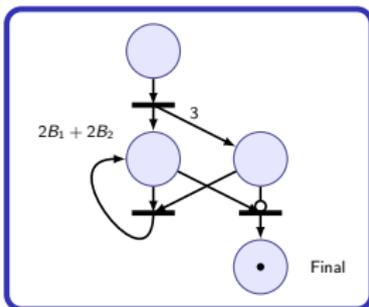
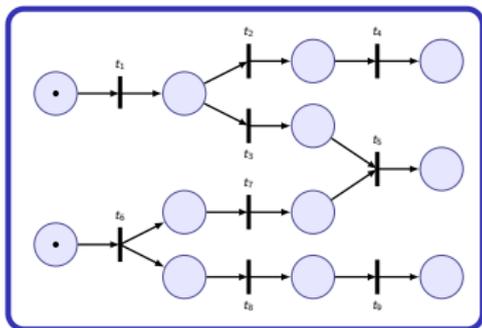


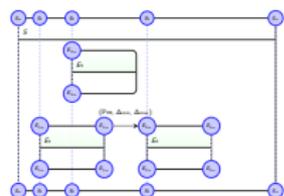
Figure: Modélisation du *Boléro* de Ravel en réseau de Petri [HR88]

## • Réseaux d'occurrences



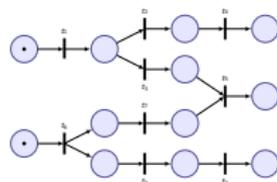
- **précédence** :  $t_1 \leq t_2, t_8 \leq t \dots$
- **concurrence** :  $t_1 || t_6, t_7 || t_8 \dots$
- **conflit** :  $t_2 \# t_3, t_2 \# t_5, t_3 \# t_4$

# Équivalences entre partitions, S-langages et réseaux d'occurrences



$\Leftrightarrow$  S-Langages

$\Leftrightarrow$



- **pécédence/posriorité :**  
 $\langle Pre, A_i, B_j, 0, \infty \rangle \Leftrightarrow \mathcal{L}(P) \times [a_i b_j]$
- **synchronisation :**  
 $\langle Pre, A_i, B_j, 0, 0 \rangle \Leftrightarrow \mathcal{L}(P) \times (a_i, b_j)$

- **concurrency :**  
 $\mathcal{L}(PN) \times [t_1 \otimes t_2]$



- **pécédence :**  
 $\mathcal{L}(PN) \times [t_1 t_2]$



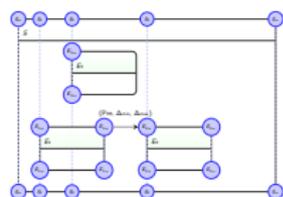
- **fusion :**  
 $\mathcal{L}(PN) \times (t_1 t_2)$



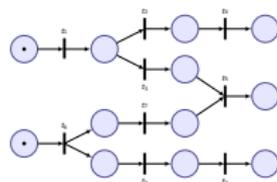
- **conflit :**  
 $\mathcal{L}(PN) \times (\mathcal{U}_1 \vee \mathcal{U}_2)$



# Équivalences entre partitions, S-langages et réseaux d'occurrences



S-Langages



- **pécédence/posriorité :**  
 $\langle Pre, A_i, B_j, 0, \infty \rangle \Leftrightarrow \mathcal{L}(P) \times [a_i b_j]$
- **synchronisation :**  
 $\langle Pre, A_i, B_j, 0, 0 \rangle \Leftrightarrow \mathcal{L}(P) \times (a_i, b_j)$

- **concurrency :**

$$\mathcal{L}(PN) \times [t_1 \otimes t_2]$$



- **pécédence :**

$$\mathcal{L}(PN) \times [t_1 t_2]$$



- **fusion :**

$$\mathcal{L}(PN) \times (t_1 t_2)$$

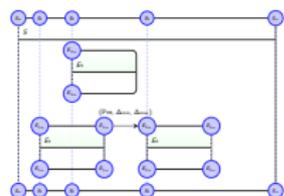


- **conflit :**

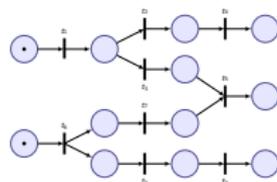
$$\mathcal{L}(PN) \times (\mathcal{U}_1 \vee \mathcal{U}_2)$$



# Équivalences entre partitions, S-langages et réseaux d'occurrences



S-Langages

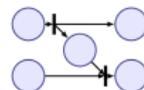


- **pécédence/posriorité :**  
 $\langle Pre, A_i, B_j, 0, \infty \rangle \Leftrightarrow \mathcal{L}(P) \bowtie [a_i b_j]$
- **synchronisation :**  
 $\langle Pre, A_i, B_j, 0, 0 \rangle \Leftrightarrow \mathcal{L}(P) \bowtie (a_i, b_j)$

- **concurrency :**  
 $\mathcal{L}(PN) \bowtie [t_1 \otimes t_2]$



- **pécédence :**  
 $\mathcal{L}(PN) \bowtie [t_1 t_2]$



- **fusion :**  
 $\mathcal{L}(PN) \bowtie (t_1 t_2)$



- **conflit :**  
 $\mathcal{L}(PN) \bowtie (\mathcal{U}_1 \vee \mathcal{U}_2)$



# Compilation

- Réseau univers pour l'alphabet des points de contrôle de  $P$  :

$$\mathcal{L}(U_P) = \mathcal{U}(\widehat{\Sigma}_P, (1, \dots, 1))$$

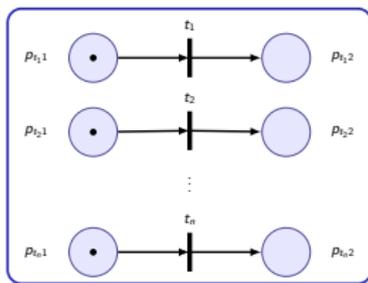


Figure: Réseau univers pour l'alphabet des points de contrôle de  $P$

- Construction du réseau d'occurrences de S-langage :

$$\mathcal{R}_{\mathcal{S}} = \mathcal{U}(\widehat{\Sigma}_P, (1, \dots, 1)) \otimes_{r \in R} s_r$$

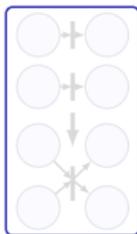


Figure: Fusion traduisant une synchronisation

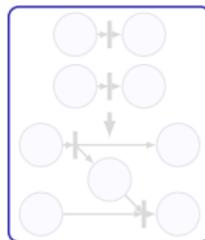


Figure: Mise en causalité traduisant une précédence

# Compilation

- Réseau univers pour l'alphabet des points de contrôle de  $P$  :

$$\mathcal{L}(U_P) = \mathcal{U}(\widehat{\Sigma}_P, (1, \dots, 1))$$

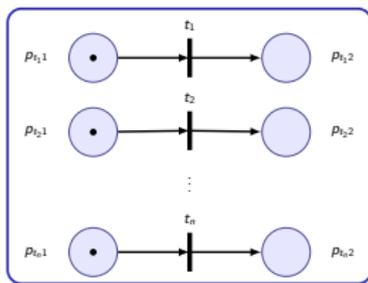


Figure: Réseau univers pour l'alphabet des points de contrôle de  $P$

- Construction du réseau d'occurrences de S-langage :

$$\mathcal{R}_{\mathcal{S}} = \mathcal{U}(\widehat{\Sigma}_P, (1, \dots, 1)) \bowtie_{r \in R} s_r$$

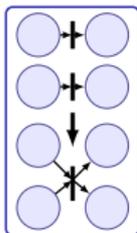


Figure: Fusion traduisant une synchronisation

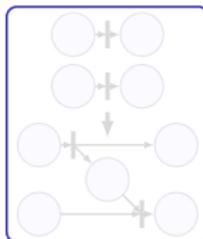


Figure: Mise en causalité traduisant une précédence

# Compilation

- Réseau univers pour l'alphabet des points de contrôle de  $P$  :

$$\mathcal{L}(U_P) = \mathcal{U}(\widehat{\Sigma}_P, (1, \dots, 1))$$

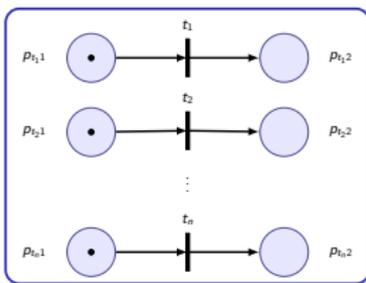


Figure: Réseau univers pour l'alphabet des points de contrôle de  $P$

- Construction du réseau d'occurrences de S-langage :

$$\mathcal{R}_{\mathcal{S}} = \mathcal{U}(\widehat{\Sigma}_P, (1, \dots, 1)) \otimes_{r \in R} s_r$$

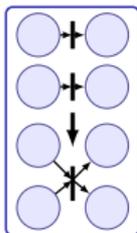


Figure: Fusion traduisant une synchronisation

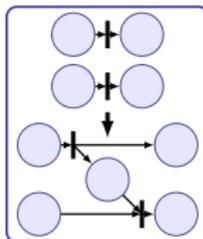


Figure: Mise en causalité traduisant une précédence

# Perspectives

- Génération de partitions interactives par raisonnement temporel avec les S-langages
  - Génération directe (utilisation des *calendriers* pour les contraintes quantitatives)
  - Application de règles d'interprétation

$$\mathcal{L}(P_{inter}) = \mathcal{L}(P_{non-inter}) \bowtie \mathcal{L}(R_{inter})$$

- Analyse



J. Haury, *La grammaire de l'exécution musicale au clavier et le mouvement des touches*, *Analyse Musicale* **7** (1987).



G. Haus and A. Rodriguez, *Music description and processing by petri nets*, *Advances on Petri Nets, Lecture Notes in Computer Science* **340** (1988), 175–199.