Control Hierarchies and Tropical Algebras

Jörg Raisch[1,2], Xavier David-Henriet[1,2,3], Tom Brunsch[1,3], Laurent Hardouin[3]

[1]Fachgebiet Regelungssysteme
Fakultät Elektrotechnik und Informatik, TU Berlin

[2]Fachgruppe System- und Regelungstheorie
Max-Planck-Institut für Dynamik komplexer technischer Systeme

[3]Laboratoire d'Ingénierie des Systèmes Automatisés
Université d'Angers

Outline

Outline

Motivation and Aims

Motivation

- Want to address large-scale/complex control problems
- Too many degrees of freedom for monolithic controller design
- Need to impose structure to reduce degrees of freedom
- Hierarchical control architecture particularly intuitive
- Heuristically designed hierarchical control ubiquituos in industry

Aims

- Want a formal framework that guarantees “proper interaction” of control layers to minimize trial and error during design
- Hierarchical structures need not be “rigid”; may be embedded into consensus-type distributed systems, with top-level functionality temporarily assigned to a node

Motivation and Aims

Motivation

- Want to address large-scale/complex control problems
- Too many degrees of freedom for monolithic controller design
- Need to impose structure to reduce degrees of freedom
- Hierarchical control architecture particularly intuitive
- Heuristically designed hierarchical control ubiquituos in industry

Aims

- Want a formal framework that guarantees “proper interaction” of control layers to minimize trial and error during design
- Hierarchical structures need not be “rigid”; may be embedded into consensus-type distributed systems, with top-level functionality temporarily assigned to a node

Outline

Abstraction and Refinement

- Have been investigated in different scenarios
- Behavioural point of view allows conceptionally (and notationally) simple explanation of main ingredients

Dynamical system with input/output structure:

$$\Sigma = \left(T, U \times Y, \mathfrak{B} \subseteq (U \times Y)^T\right)$$

Abstractions and refinements:

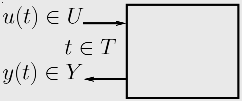
- $\Sigma_a = (T, U \times Y, \mathfrak{B}_a)$ is an abstraction of Σ if $\mathfrak{B} \subseteq \mathfrak{B}_a$
- $\Sigma_r = (T, U \times Y, \mathfrak{B}_r)$ is a refinement of Σ if $\mathfrak{B}_r \subseteq \mathfrak{B}$

Interpretation: abstraction (refinement) corresponds to adding (removing) uncertainty

Abstraction and Refinement

- Have been investigated in different scenarios
- Behavioural point of view allows conceptionally (and notationally) simple explanation of main ingredients

Dynamical system with input/output structure:



$$\Sigma = \left(T, U \times Y, \mathfrak{B} \subseteq (U \times Y)^T\right)$$

Abstractions and refinements:

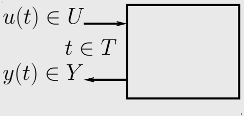
- $\Sigma_a = (T, U \times Y, \mathfrak{B}_a)$ is an abstraction of Σ if $\mathfrak{B} \subseteq \mathfrak{B}_a$
- $\Sigma_r = (T, U \times Y, \mathfrak{B}_r)$ is a refinement of Σ if $\mathfrak{B}_r \subseteq \mathfrak{B}$

Interpretation: abstraction (refinement) corresponds to adding (removing) uncertainty

Abstraction and Refinement

- Have been investigated in different scenarios
- Behavioural point of view allows conceptionally (and notationally) simple explanation of main ingredients

Dynamical system with input/output structure:



$$\Sigma = \left(T, U \times Y, \mathfrak{B} \subseteq (U \times Y)^T\right)$$

Abstractions and refinements:

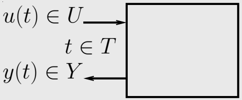
- $\Sigma_a = (T, U \times Y, \mathfrak{B}_a)$ is an abstraction of Σ if $\mathfrak{B} \subseteq \mathfrak{B}_a$
- $\Sigma_r = (T, U \times Y, \mathfrak{B}_r)$ is a refinement of Σ if $\mathfrak{B}_r \subseteq \mathfrak{B}$

Interpretation: abstraction (refinement) corresponds to adding (removing) uncertainty

Abstraction and Refinement

- Have been investigated in different scenarios
- Behavioural point of view allows conceptionally (and notationally) simple explanation of main ingredients

Dynamical system with input/output structure:



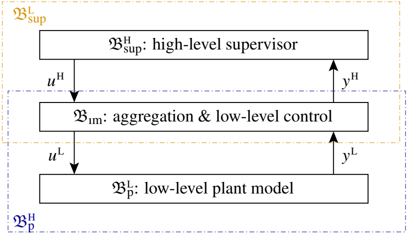
$$\Sigma = \left(T, U \times Y, \mathfrak{B} \subseteq (U \times Y)^T\right)$$

Abstractions and refinements:

- $\Sigma_a = (T, U \times Y, \mathfrak{B}_a)$ is an abstraction of Σ if $\mathfrak{B} \subseteq \mathfrak{B}_a$
- $\Sigma_r = (T, U \times Y, \mathfrak{B}_r)$ is a refinement of Σ if $\mathfrak{B}_r \subseteq \mathfrak{B}$

Interpretation: abstraction (refinement) corresponds to adding (removing) uncertainty

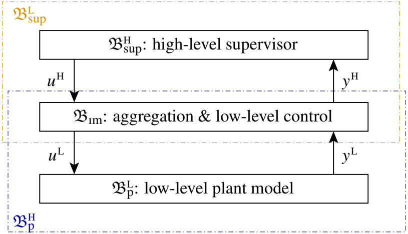
Generic Two-Level Control Structure



. . . can be extended to arbitrary number of control layers . . .

- Low-level signal space: $W_{\mathrm{L}} = U_{\mathrm{L}} \times Y_{\mathrm{L}}$.
- Low-level process model: $\mathfrak{B}^{\mathrm{L}}_{\mathrm{p}}$ . . . behaviour on W_{L}.
- Inclusion-type specification: $\mathfrak{B}^{\mathrm{L}}_{\mathrm{spec}}$ . . . defined on W_{L}.
- High-level signal space: $W_{\mathrm{H}} = U_{\mathrm{H}} \times Y_{\mathrm{H}}$.
- High-level supervisor: $\mathfrak{B}^{\mathrm{H}}_{\mathrm{sup}}$ . . . behaviour on W_{H}.
- Low-level control: $\mathfrak{B}_{\mathrm{lm}}$ . . . behaviour on $W_{\mathrm{H}} \times W_{\mathrm{L}}$.

Generic Two-Level Control Structure



…can be extended to arbitrary number of control layers …

- Low-level signal space: $W_L = U_L \times Y_L$.
- Low-level process model: $\mathfrak{B}^L_p$ …behaviour on W_L.
- Inclusion-type specification: $\mathfrak{B}^L_{spec}$ …defined on W_L.
- High-level signal space: $W_H = U_H \times Y_H$.
- High-level supervisor: $\mathfrak{B}^H_{sup}$ …behaviour on W_H.
- Low-level control: $\mathfrak{B}_{IM}$ …behaviour on $W_H \times W_L$.

Design Procedure

Define high-level signal space (assumed given in this talk).

Low-level control:

- Define (inclusion-type) specs $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ for lower control layer – *intended* relation between high-level and low-level signals.
- Design low-level control $\mathfrak{B}_{\mathsf{im}}$ enforcing specs $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$.

High-level control:

Synthesise $\mathfrak{B}^{\mathrm{H}}_{\mathrm{sup}}$ for $\mathfrak{B}^{\mathrm{H}}_{\mathrm{p}} = \mathfrak{B}^{\mathrm{H}}_{\mathsf{im}}[\mathfrak{B}^{\mathrm{L}}_{\mathrm{p}}]$. Can be done abstraction-based!

- Use high-level proj. $\mathrm{P}^{\mathrm{H}}(\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}})$ of $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ as abstraction of $\mathfrak{B}^{\mathrm{H}}_{\mathrm{p}}$.
- Define high-level spec. $\mathfrak{B}^{\mathrm{H}}_{\mathrm{spec}}$ such that $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}[\mathfrak{B}^{\mathrm{H}}_{\mathrm{spec}}] \subseteq \mathfrak{B}^{\mathrm{L}}_{\mathrm{spec}}$.
- Find high-level control $\mathfrak{B}^{\mathrm{H}}_{\mathrm{sup}}$ such that $\mathrm{P}^{\mathrm{H}}(\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}) \cap \mathfrak{B}^{\mathrm{H}}_{\mathrm{sup}} \subseteq \mathfrak{B}^{\mathrm{H}}_{\mathrm{spec}}$.

$$\Longrightarrow \mathfrak{B}^{\mathrm{L}}_{\mathrm{p}} \cap \underbrace{\mathfrak{B}^{\mathrm{L}}_{\mathsf{im}}[\mathfrak{B}^{\mathrm{H}}_{\mathrm{sup}}]}_{\mathfrak{B}^{\mathrm{L}}_{\mathrm{sup}}} \subseteq \mathfrak{B}^{\mathrm{L}}_{\mathrm{spec}}$$

Design Procedure

Define high-level signal space (assumed given in this talk).

Low-level control:

- Define (inclusion-type) specs $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ for lower control layer – *intended* relation between high-level and low-level signals.
- Design low-level control $\mathfrak{B}_{\mathrm{im}}$ enforcing specs $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$.

High-level control:

Synthesise $\mathfrak{B}^{\mathrm{H}}_{\mathrm{sup}}$ for $\mathfrak{B}^{\mathrm{H}}_{\mathrm{p}} = \mathfrak{B}^{\mathrm{H}}_{\mathrm{im}}[\mathfrak{B}^{\mathrm{L}}_{\mathrm{p}}]$. Can be done abstraction-based!

- Use high-level proj. $\mathrm{P}^{\mathrm{H}}(\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}})$ of $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ as abstraction of $\mathfrak{B}^{\mathrm{H}}_{\mathrm{p}}$.
- Define high-level spec. $\mathfrak{B}^{\mathrm{H}}_{\mathrm{spec}}$ such that $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}[\mathfrak{B}^{\mathrm{H}}_{\mathrm{spec}}] \subseteq \mathfrak{B}^{\mathrm{L}}_{\mathrm{spec}}$.
- Find high-level control $\mathfrak{B}^{\mathrm{H}}_{\mathrm{sup}}$ such that $\mathrm{P}^{\mathrm{H}}(\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}) \cap \mathfrak{B}^{\mathrm{H}}_{\mathrm{sup}} \subseteq \mathfrak{B}^{\mathrm{H}}_{\mathrm{spec}}$.

$$\Longrightarrow \mathfrak{B}^{\mathrm{L}}_{\mathrm{p}} \cap \underbrace{\mathfrak{B}^{\mathrm{L}}_{\mathrm{im}}[\mathfrak{B}^{\mathrm{H}}_{\mathrm{sup}}]}_{\mathfrak{B}^{\mathrm{L}}_{\mathrm{sup}}} \subseteq \mathfrak{B}^{\mathrm{L}}_{\mathrm{spec}}$$

Design Procedure

Define high-level signal space (assumed given in this talk).

Low-level control:

- Define (inclusion-type) specs $\mathfrak{B}^{HL}_{spec}$ for lower control layer – *intended* relation between high-level and low-level signals.
- Design low-level control $\mathfrak{B}_{im}$ enforcing specs $\mathfrak{B}^{HL}_{spec}$.

High-level control:

Synthesise $\mathfrak{B}^{H}_{sup}$ for $\mathfrak{B}^{H}_{p} = \mathfrak{B}^{H}_{im}[\mathfrak{B}^{L}_{p}]$. Can be done abstraction-based!

- Use high-level proj. $P^H(\mathfrak{B}^{HL}_{spec})$ of $\mathfrak{B}^{HL}_{spec}$ as abstraction of $\mathfrak{B}^{H}_{p}$.
- Define high-level spec. $\mathfrak{B}^{H}_{spec}$ such that $\mathfrak{B}^{HL}_{spec}[\mathfrak{B}^{H}_{spec}] \subseteq \mathfrak{B}^{L}_{spec}$.
- Find high-level control $\mathfrak{B}^{H}_{sup}$ such that $P^H(\mathfrak{B}^{HL}_{spec}) \cap \mathfrak{B}^{H}_{sup} \subseteq \mathfrak{B}^{H}_{spec}$.

$$\Longrightarrow \mathfrak{B}^{L}_{p} \cap \underbrace{\mathfrak{B}^{L}_{im}[\mathfrak{B}^{H}_{sup}]}_{\mathfrak{B}^{L}_{sup}} \subseteq \mathfrak{B}^{L}_{spec}$$

Where Can Things Go Wrong?

Low-level specification $\mathfrak{B}_{\text{spec}}^{\text{HL}}$ too demanding:

- I.e., we cannot find appropriate low-level control.
- Need to relax low-level specifications and replace $\mathfrak{B}_{\text{spec}}^{\text{HL}}$ by an abstraction $\mathfrak{B}_{\text{spec,a}}^{\text{HL}}$ such that $\mathfrak{B}_{\text{spec}}^{\text{HL}} \subseteq \mathfrak{B}_{\text{spec,a}}^{\text{HL}}$.

Illustration: robot moving in a restricted area:

$$\begin{aligned} \dot{x}_1(t) &= v(t)\cos\theta(t) \\ \dot{x}_2(t) &= v(t)\sin\theta(t) \\ \dot{\theta}(t) &= u_1(t) \\ \dot{v}(t) &= u_2(t) \end{aligned}$$

$u^{\text{L}} = (u_1, u_2)$ low-level inputs
$y^{\text{L}} = (x_1, x_2)$ low-level outputs
$u^{\text{H}} \in \{\text{go up}, \ldots\}$ high-level input
$y^{\text{H}} = \text{quant}(x_1, x_2)$ high-lev. outp.

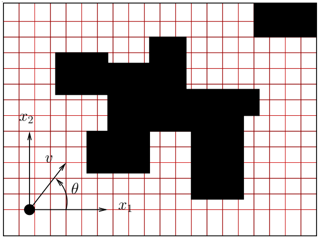
Where Can Things Go Wrong?

Low-level specification $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ too demanding:

- I.e., we cannot find appropriate low-level control.
- Need to relax low-level specifications and replace $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ by an abstraction $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec,a}}$ such that $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}} \subseteq \mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec,a}}$.

Illustration: robot moving in a restricted area:

$$\begin{aligned}\dot{x}_1(t) &= v(t)\cos\theta(t)\\ \dot{x}_2(t) &= v(t)\sin\theta(t)\\ \dot{\theta}(t) &= u_1(t)\\ \dot{v}(t) &= u_2(t)\end{aligned}$$



$u^{\mathrm{L}} = (u_1, u_2)$ low-level inputs
$y^{\mathrm{L}} = (x_1, x_2)$ low-level outputs
$u^{\mathrm{H}} \in \{\text{go up}, \ldots\}$ high-level input
$y^{\mathrm{H}} = \text{quant}(x_1, x_2)$ high-lev. outp.

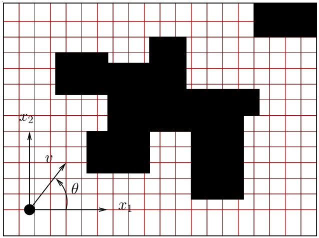
Where Can Things Go Wrong?

Low-level specification $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ too demanding:

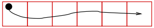
- I.e., we cannot find appropriate low-level control.
- Need to relax low-level specifications and replace $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ by an abstraction $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec,a}}$ such that $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}} \subseteq \mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec,a}}$.

Illustration: robot moving in a restricted area:

$$
\begin{aligned}
\dot{x}_1(t) &= v(t)\cos\theta(t)\\
\dot{x}_2(t) &= v(t)\sin\theta(t)\\
\dot{\theta}(t) &= u_1(t)\\
\dot{v}(t) &= u_2(t)
\end{aligned}
$$



$u^{\mathrm{L}} = (u_1, u_2)$ low-level inputs
$y^{\mathrm{L}} = (x_1, x_2)$ low-level outputs
$u^{\mathrm{H}} \in \{\text{go up}, \ldots\}$ high-level input
$y^{\mathrm{H}} = \mathrm{quant}(x_1, x_2)$ high-lev. outp.

Ex.: "go right" $\rightsquigarrow$ 
is too demanding.

What Else Can Go Wrong?

Low-level specification $\mathfrak{B}^{\text{HL}}_{\text{spec}}$ too coarse:

- $P^{\text{H}}(\mathfrak{B}^{\text{HL}}_{\text{spec}})$ serves as abstraction of plant under low-level control.
- We cannot find appropriate high-level control.
- Need to refine low-level specifications by $\mathfrak{B}^{\text{HL}}_{\text{spec,r}} \subseteq \mathfrak{B}^{\text{HL}}_{\text{spec}}$.

Example:

Recap:

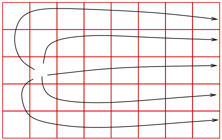
- choice of low-level specs $\mathfrak{B}^{\text{HL}}_{\text{spec}}$ depends on engineering intuition
- often involves trade-off between control layers
- key advantage: solution of low- & high-level control problems will provide a solution for the overall problem (guaranteed!)

What Else Can Go Wrong?

Low-level specification $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ too coarse:

- $P^{\mathrm{H}}(\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}})$ serves as abstraction of plant under low-level control.
- We cannot find appropriate high-level control.
- Need to refine low-level specifications by $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec,r}} \subseteq \mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$.

Example: "go right" $\rightsquigarrow$



Recap:

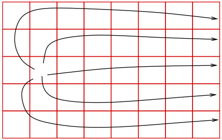
- choice of low-level specs $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ depends on engineering intuition
- often involves trade-off between control layers
- key advantage: solution of low- & high-level control problems will provide a solution for the overall problem (guaranteed!)

What Else Can Go Wrong?

Low-level specification $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ too coarse:

- $\mathrm{P}^{\mathrm{H}}(\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}})$ serves as abstraction of plant under low-level control.
- We cannot find appropriate high-level control.
- Need to refine low-level specifications by $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec,r}} \subseteq \mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$.

Example: "go right" $\rightsquigarrow$



Recap:

- choice of low-level specs $\mathfrak{B}^{\mathrm{HL}}_{\mathrm{spec}}$ depends on engineering intuition
- often involves trade-off between control layers
- key advantage: solution of low- & high-level control problems will provide a solution for the overall problem (guaranteed!)

Outline

Specific Scenario

- top layer decides on timing (not ordering!) of discrete events
- synthesis based on TEG abstraction of plant + low-level control
- TEG (Timed Event Graph) . . . specific timed Petri net

Example:

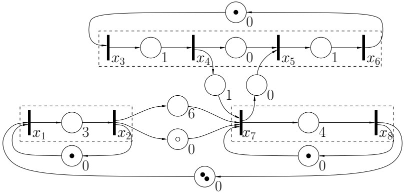
$$x_7(k) = \max\{x_4(k) + 1, x_2(k) + 6, x_2(k+1), x_8(k-1)\}$$

time relations become linear in certain dioid (tropical) algebras

Specific Scenario

- top layer decides on timing (not ordering!) of discrete events
- synthesis based on TEG abstraction of plant + low-level control
- TEG (Timed Event Graph) . . . specific timed Petri net

Example:



- want to compute earliest times of k-th occurrences of events
- doable, but time relations (non-benevolently) non-linear

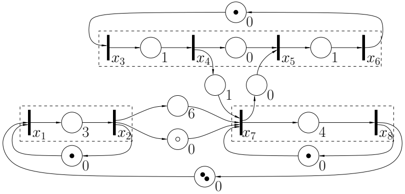
$$x_7(k) = \max\{x_4(k) + 1, x_2(k) + 6, x_2(k+1), x_8(k-1)\}$$

time relations become linear in certain dioid (tropical) algebras . . .

Specific Scenario

- top layer decides on timing (not ordering!) of discrete events
- synthesis based on TEG abstraction of plant + low-level control
- TEG (Timed Event Graph) . . . specific timed Petri net

Example:



- want to compute earliest times of k-th occurrences of events
- doable, but time relations (non-benevolently) non-linear

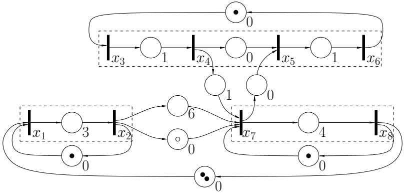
$$x_7(k) = \max\{x_4(k) + 1, x_2(k) + 6, x_2(k+1), x_8(k-1)\}$$

time relations become linear in certain dioid (tropical) algebras . . .

Specific Scenario

- top layer decides on timing (not ordering!) of discrete events
- synthesis based on TEG abstraction of plant + low-level control
- TEG (Timed Event Graph) . . . specific timed Petri net

Example:



- want to compute earliest times of k-th occurrences of events
- doable, but time relations (non-benevolently) non-linear

$$x_7(k) = \max\{x_4(k) + 1, x_2(k) + 6, x_2(k+1), x_8(k-1)\}$$

time relations become linear in certain dioid (tropical) algebras . . .

Outline

Dioid Algebras

A dioid is an algebraic structure with two binary operations $\oplus$ ("addition") and $\otimes$ ("multiplication") defined on a set $\mathcal{D}$, such that

- $\oplus$ is associative, commutative & idempotent ($a \oplus a = a \; \forall a \in \mathcal{D}$)
- $\otimes$ is associative and is distributive w.r.t. $\oplus$
- zero element ε, unit element e
- ε is absorbing for $\otimes$, i.e., $\varepsilon \otimes a = a \otimes \varepsilon = \varepsilon \; \forall a \in \mathcal{D}$

Remarks

- a dioid is complete if it is closed for infinite sums and $\otimes$ distributes over infinite sums
- dioids are equipped with a natural order: $a \oplus b = a \Leftrightarrow a \succeq b$
- addition and multiplication can be easiliy extended to matrices

Dioid Algebras

A dioid is an algebraic structure with two binary operations $\oplus$ ("addition") and $\otimes$ ("multiplication") defined on a set $\mathcal{D}$, such that

- $\oplus$ is associative, commutative & idempotent ($a \oplus a = a \, \forall a \in \mathcal{D}$)
- $\otimes$ is associative and is distributive w.r.t. $\oplus$
- zero element ε, unit element e
- ε is absorbing for $\otimes$, i.e., $\varepsilon \otimes a = a \otimes \varepsilon = \varepsilon \, \forall a \in \mathcal{D}$

Remarks

- a dioid is complete if it is closed for infinite sums and $\otimes$ distributes over infinite sums
- dioids are equipped with a natural order: $a \oplus b = a \Leftrightarrow a \succeq b$
- addition and multiplication can be easiliy extended to matrices

Example: The Max-Plus Algebra

Defined on $\overline{\mathbb{Z}} = \mathbb{Z} \cup \{-\infty\} \cup \{+\infty\}$ resp. $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty\} \cup \{+\infty\}$:

- addition: $a \oplus b := \max(a, b)$, zero element: $\varepsilon := -\infty$
- multiplication: $a \otimes b := a + b$, unit element: $e := 0$

Time relations for TEGs described by linear implicit difference eqns.

For our example

$$x_7(k) = 1 \otimes x_4(k) \oplus 6 \otimes x_2(k) \oplus x_2(k+1) \oplus x_8(k-1)$$

Example: The Max-Plus Algebra

Defined on $\overline{\mathbb{Z}} = \mathbb{Z} \cup \{-\infty\} \cup \{+\infty\}$ resp. $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty\} \cup \{+\infty\}$:

- addition: $a \oplus b := \max(a, b)$, zero element: $\varepsilon := -\infty$
- multiplication: $a \otimes b := a + b$, unit element: $e := 0$

Time relations for TEGs described by linear implicit difference eqns.

For our example

$$x_7(k) = 1 \otimes x_4(k) \oplus 6 \otimes x_2(k) \oplus x_2(k+1) \oplus x_8(k-1)$$

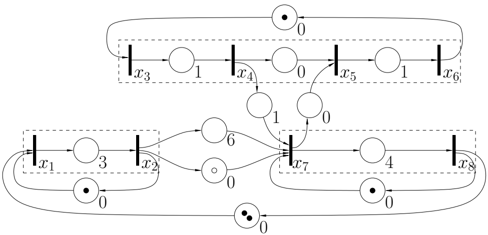
Example: The Max-Plus Algebra

Defined on $\overline{\mathbb{Z}} = \mathbb{Z} \cup \{-\infty\} \cup \{+\infty\}$ resp. $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty\} \cup \{+\infty\}$:

- addition: $a \oplus b := \max(a, b)$, zero element: $\varepsilon := -\infty$
- multiplication: $a \otimes b := a + b$, unit element: $e := 0$

Time relations for TEGs described by linear implicit difference eqns.

For our example



$$x_7(k) = 1 \otimes x_4(k) \oplus 6 \otimes x_2(k) \oplus x_2(k+1) \oplus x_8(k-1)$$

The Dioid $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$

- $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ ... a quotient dioid in the set of 2-dim. formal power series (in γ, δ), with Boolean coefficients and integer exponents
- interpretation of monomial $\gamma^k\delta^t$:
  - kth occurrence of event is at time t at the earliest
  - equivalently: at time t, event has occurred at most k times

$\rightsquigarrow$ have to consider "south-east cones" (instead of points) in $\overline{\mathbb{Z}}^2$

Example: $s = \gamma^1\delta^1 \oplus \gamma^3\delta^2 \oplus \gamma^4\delta^5$

The Dioid $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$

- $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ . . . a quotient dioid in the set of 2-dim. formal power series (in γ, δ), with Boolean coefficients and integer exponents
- interpretation of monomial $\gamma^k\delta^t$:
  - *k*th occurrence of event is at time *t* at the earliest
  - equivalently: at time *t*, event has occurred at most *k* times

  $\rightsquigarrow$ have to consider "south-east cones" (instead of points) in $\overline{\mathbb{Z}}^2$

Example: $s = \gamma^1\delta^1 \oplus \gamma^3\delta^2 \oplus \gamma^4\delta^5$

Properties:

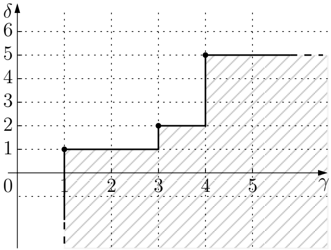
- $\gamma^k\delta^t \oplus \gamma^l\delta^t = \gamma^{\min(k,l)}\delta^t$
- $\gamma^k\delta^t \oplus \gamma^k\delta^\tau = \gamma^k\delta^{\max(t,\tau)}$
- $\gamma^k\delta^t \otimes \gamma^l\delta^\tau = \gamma^{(k+l)}\delta^{(t+\tau)}$
- Zero element: $\varepsilon = \gamma^{+\infty}\delta^{-\infty}$
- Unit element: $e = \gamma^0\delta^0$
- interpretation of partial order: inclusion in $\overline{\mathbb{Z}}^2$

The Dioid $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$

- $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ ... a quotient dioid in the set of 2-dim. formal power series (in γ, δ), with Boolean coefficients and integer exponents
- interpretation of monomial $\gamma^k\delta^t$:
  - *k*th occurrence of event is at time *t* at the earliest
  - equivalently: at time *t*, event has occurred at most *k* times

$\rightsquigarrow$ have to consider "south-east cones" (instead of points) in $\overline{\mathbb{Z}}^2$

Example: $s = \gamma^1\delta^1 \oplus \gamma^3\delta^2 \oplus \gamma^4\delta^5$



Properties:

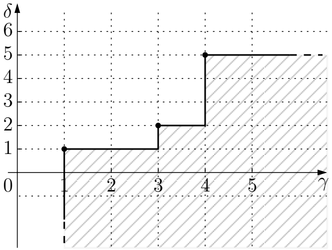
- $\gamma^k\delta^t \oplus \gamma^l\delta^t = \gamma^{\min(k,l)}\delta^t$
- $\gamma^k\delta^t \oplus \gamma^k\delta^\tau = \gamma^k\delta^{\max(t,\tau)}$
- $\gamma^k\delta^t \otimes \gamma^l\delta^\tau = \gamma^{(k+l)}\delta^{(t+\tau)}$
- Zero element: $\varepsilon = \gamma^{+\infty}\delta^{-\infty}$
- Unit element: $e = \gamma^0\delta^0$
- interpretation of partial order: inclusion in $\overline{\mathbb{Z}}^2$

The Dioid $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$

- $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$ ... a quotient dioid in the set of 2-dim. formal power series (in γ, δ), with Boolean coefficients and integer exponents
- interpretation of monomial $\gamma^k \delta^t$:
  - *k*th occurrence of event is at time *t* at the earliest
  - equivalently: at time *t*, event has occurred at most *k* times

$\rightsquigarrow$ have to consider “south-east cones” (instead of points) in $\overline{\mathbb{Z}}^2$

Example: $s = \gamma^1\delta^1 \oplus \gamma^3\delta^2 \oplus \gamma^4\delta^5$



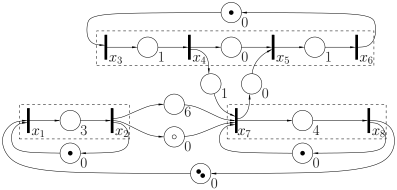
Properties:

- $\gamma^k\delta^t \oplus \gamma^l\delta^t = \gamma^{\min(k,l)}\delta^t$
- $\gamma^k\delta^t \oplus \gamma^k\delta^\tau = \gamma^k\delta^{\max(t,\tau)}$
- $\gamma^k\delta^t \otimes \gamma^l\delta^\tau = \gamma^{(k+l)}\delta^{(t+\tau)}$
- Zero element: $\varepsilon = \gamma^{+\infty}\delta^{-\infty}$
- Unit element: $e = \gamma^0\delta^0$
- interpretation of partial order: inclusion in $\overline{\mathbb{Z}}^2$

The Dioid $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ ctd.

Time relations for TEGs become linear algebraic eqns. in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$

For our example



$$x_7 = \delta^1\gamma^0 x_4 \oplus (\delta^6\gamma^0 \oplus \delta^0\gamma^{-1})x_2 \oplus \delta^0\gamma^1 x_8$$

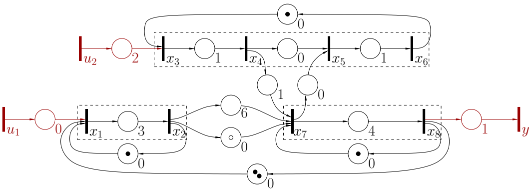
In general, with input & output trans. (triggered resp. seen externally):

$$\begin{aligned} x &= Ax \oplus Bu \\ y &= Cx \end{aligned}$$

The Dioid $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$ ctd.

Time relations for TEGs become linear algebraic eqns. in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$

For our example



$$x_7 = \delta^1\gamma^0 x_4 \oplus (\delta^6\gamma^0 \oplus \delta^0\gamma^{-1})x_2 \oplus \delta^0\gamma^1 x_8$$

In general, with input & output trans. (triggered resp. seen externally):

$$\begin{aligned} x &= Ax \oplus Bu \\ y &= Cx \end{aligned}$$

Control in the Dioid $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$

Plant:

- state model $x = Ax \oplus Bu,\ y = Cx$
- i/o rel. $y = CA^*Bu$, with $A^* := \bigoplus_{i\in\mathbb{N}_0} A^i$ . . . Kleene star operator

Output feedback:

$$\begin{aligned} u &= Ky \oplus v \\ \rightsquigarrow y &= CA^*BKy \oplus CA^*Bv \\ y &= \underbrace{(CA^*BK)^*CA^*B}_{H_{cl}}\, v \end{aligned}$$

Aim: just-in-time policy

find greatest K s.t. $H_{ref} \succeq H_{cl}$, with

- H_{ref} a given reference model
- "greatest" and "$\succeq$" in the sense of natural order in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$

Solution:

desired feedback K can be obtained using "residuation theory":

$$K_{opt} = (CA^*B) \backslash\!\!\circ H_{ref} \not/ (CA^*B)$$

Control in the Dioid $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$

Plant:

- state model $x = Ax \oplus Bu,\ \ y = Cx$
- i/o rel. $y = CA^* Bu$, with $A^* := \bigoplus_{i \in \mathbb{N}_0} A^i$ . . . Kleene star operator

Output feedback:

$$\begin{aligned} u &= Ky \oplus v \\ \rightsquigarrow y &= CA^* BKy \oplus CA^* Bv \\ y &= \underbrace{(CA^* BK)^* CA^* B}_{H_{cl}}\, v \end{aligned}$$

Aim: just-in-time policy

find greatest K s.t. $H_{ref} \succeq H_{cl}$, with

- H_{ref} a given reference model
- "greatest" and "$\succeq$" in the sense of natural order in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$

Solution:

desired feedback K can be obtained using "residuation theory":

$$K_{opt} = (CA^* B) \backslash\!\!\circ H_{ref} \,/\!\!\circ\, (CA^* B)$$

Control in the Dioid $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$

Plant:

- state model $x = Ax \oplus Bu,\ y = Cx$
- i/o rel. $y = CA^*Bu$, with $A^* := \bigoplus_{i\in\mathbb{N}_0} A^i$ ... Kleene star operator

Output feedback:

$$\begin{aligned} u &= Ky \oplus v \\ \rightsquigarrow y &= CA^*BKy \oplus CA^*Bv \\ y &= \underbrace{(CA^*BK)^*CA^*B}_{H_{cl}}\, v \end{aligned}$$

Aim: just-in-time policy

find greatest K s.t. $H_{ref} \succeq H_{cl}$, with

- H_{ref} a given reference model
- "greatest" and "$\succeq$" in the sense of natural order in $\mathcal{M}_{in}^{ax}[\![\gamma,\delta]\!]$

Solution:

desired feedback K can be obtained using "residuation theory":

$$K_{opt} = (CA^*B) \backslash\!\!\circ H_{ref} \,/\!\!\circ\, (CA^*B)$$

Control in the Dioid $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$

Plant:

- state model $x = Ax \oplus Bu, \; y = Cx$
- i/o rel. $y = CA^*Bu$, with $A^* := \bigoplus_{i \in \mathbb{N}_0} A^i$ ...Kleene star operator

Output feedback:

$$u = Ky \oplus v$$
$$\rightsquigarrow y = CA^*BKy \oplus CA^*Bv$$
$$y = \underbrace{(CA^*BK)^* CA^*B}_{H_{cl}} v$$

Aim: just-in-time policy

find greatest K s.t. $H_{ref} \succeq H_{cl}$, with

- H_{ref} a given reference model
- "greatest" and "$\succeq$" in the sense of natural order in $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$

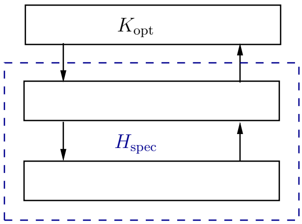
Solution:

desired feedback K can be obtained using "residuation theory":

$$K_{opt} = (CA^*B) \circ\!\!\!\backslash H_{ref} \circ\!\!\!/ (CA^*B)$$

Outline

Tradeoff Between Control Layers



- K_{opt} ... greatest feedback K s.t.

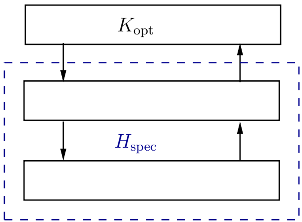
$$(H_{\text{spec}} K)^* H_{\text{spec}} \preceq G_{\text{spec}}$$

  for a given overall spec. G_{spec}
- H_{spec} ... low-level spec., i.e., abstraction for plant under low-level control

Result:

- Given overall specification G_{spec}
- Given low-level specifications H_{spec_1}, H_{spec_2}, with $H_{\text{spec}_1} \preceq H_{\text{spec}_2}$ (and some "natural" restrictions in place)
- Compute corresponding optimal feedback control K_{opt_1}, K_{opt_2}
- Can show that $K_{\text{opt}_1} \succeq K_{\text{opt}_2}$ ("stricter low-level specs allow for more relaxed high-level control")

Tradeoff Between Control Layers



- K_{opt} ... greatest feedback K s.t.

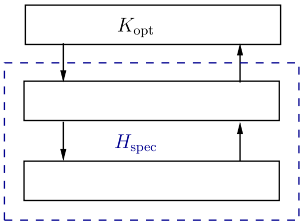
$$(H_{\text{spec}}K)^* H_{\text{spec}} \preceq G_{\text{spec}}$$

  for a given overall spec. G_{spec}
- H_{spec} ... low-level spec., i.e., abstraction for plant under low-level control

Result:

- Given overall specification G_{spec}
- Given low-level specifications H_{spec_1}, H_{spec_2}, with $H_{\text{spec}_1} \preceq H_{\text{spec}_2}$ (and some “natural” restrictions in place)
- Compute corresponding optimal feedback control K_{opt_1}, K_{opt_2}
- Can show that $K_{\text{opt}_1} \succeq K_{\text{opt}_2}$ (“stricter low-level specs allow for more relaxed high-level control”)

Tradeoff Between Control Layers



- K_{opt} ... greatest feedback K s.t.

$$(H_{\text{spec}}K)^* H_{\text{spec}} \preceq G_{\text{spec}}$$

  for a given overall spec. G_{spec}
- H_{spec} ... low-level spec., i.e., abstraction for plant under low-level control

Result:

- Given overall specification G_{spec}
- Given low-level specifications H_{spec_1}, H_{spec_2}, with $H_{\text{spec}_1} \preceq H_{\text{spec}_2}$ (and some "natural" restrictions in place)
- Compute corresponding optimal feedback control K_{opt_1}, K_{opt_2}
- Can show that $K_{\text{opt}_1} \succeq K_{\text{opt}_2}$ ("stricter low-level specs allow for more relaxed high-level control")

Conclusions

- Interpreted trade-off between layers in a hierarchical control system from a behavioural point of view
- Formally investigated this trade-off for a specific scenario where top layer is responsible for timing of discrete events
- Resulting setup conveniently described in the dioid $\mathcal{M}_{in}^{ax}[\![\gamma, \delta]\!]$
- Verified that stricter low-level specs indeed allow for more relaxed high-level control

More Details

[1] J. Willems: Models for dynamics, in *Dynamics Reported*, vol. 2, 1989, pp. 172–269.

[2] J. Raisch and T. Moor: Hierarchical Hybrid Control of a Multiproduct Batch Plant, *Lecture Notes in Control and Information Sciences.* Springer-Verlag, 2005, vol. 322, p. 199–216.

[3] F. Baccelli, G. Cohen, G.J. Olsder, and J.-P. Quadrat: Synchronization and Linearity – An Algebra for Discrete Event Systems. Wiley, 1992.

[4] B. Cottenceau, L. Hardouin, J.-L. Boimond, and J.-L. Ferrier: Model reference control for timed event graphs in dioids, *Automatica*, vol. 37, no. 9, pp. 1451–1458, 2001.

[5] M. Lhommeau, L. Hardouin, R. Santos Mendes and B. Cottenceau: On the model reference control for max-plus linear systems. *Proc. 44th IEEE CDC*, pp. 7799–7803, 2005.

More Details (ctd.)

[6] X. David-Henriet, J. Raisch, and L. Hardouin: Consistent control hierarchies with top layers represented by timed event graphs. *Proc. MMAR 2012 – 17th Int. Conf. on Methods and Models in Automation and Robotics*, 2012.

[7] T. Brunsch, L. Hardouin, C. A. Maia and J. Raisch. Duality and interval analysis over idempotent semirings, *Linear Algebra and its Applications*, vol. 437, no. 10, pp. 2436–2454, 2012.

[8] T. Brunsch, J. Raisch, L. Hardouin and O. Boutin. Discrete-Event Systems in a Dioid Framework: Modeling and Analysis, *Lecture Notes in Control and Information Sciences*, vol. 433, pp. 431–450, Springer-Verlag, 2012.

[9] L. Hardouin, O. Boutin, B. Cottenceau, T. Brunsch, and J. Raisch. Discrete-Event Systems in a Dioid Framework: Control Theory, *Lecture Notes in Control and Information Sciences*, vol. 433, pp. 451–469, Springer-Verlag, 2012.