Requirements on the rule base [Moser, Navara 2002]

- **Local correctness (interaction)**: $\forall j : \Phi(A_j) = C_j$.
- **Strong completeness**: $\forall$ normal $X \in \mathcal{F}(\mathcal{X}) : \Phi(X) \not\subseteq \bigcap_i C_i$, where the fuzzy intersection is standard (computed using $\min$).
- **Weak interpolation property**: $\Phi(X)$ is in the convex hull of all C_i with i such that Supp $A_i \cap$ Supp $X \neq \emptyset$.
- **Crisp correctness (crisp interaction)**: $\big(A_i(x) = 1\big) \Rightarrow \big(\Phi(x) = \Phi(\{x\}) = C_i\big)$ ("if there is a totally firing rule, it determines the output").

Completeness of the rule base

Completeness is required, because in any situation we need at least one firing rule.

Nevertheless, non-completeness is sometimes tolerated for the following reasons:

- In expert systems; "I don't know" could be a legitimate answer (of an expert system, not of a pilot!).
- The input is impossible (then do not include it in the input space!).
- The input values are fuzzified so that they always overlap with some antecedent.
- The sparse database is used for interpolation [Kóczy et al. 1997].
- Some inputs do not require any action (we just wait until the situation changes).

The latter case can be formally described by an additional "**else rule**"
[Amato, Di Nola, Navara 2003].
It is treated differently w.r.t. other requirements.
In any case, it assumes that we assign a meaning of "no action".
the output variable has to be defined always.

Completeness of the rule base

Omitting rules for some situations is motivated by the attempt to reduce the number of rules (**curse of dimensionality**).

Sometimes the table of linguistic rules does not cover some combinations of linguistic variables.
This does not obviously mean that the antecedents are not complete; the case may be covered by neighbouring rules, although with a smaller degree of firing.

Correctness of Mamdani–Assilian controller

When $\forall j : \Phi(A_j) = A_j \underset{\cdot}{\circ} R_{\textsf{MA}} = C_j$? (A system of fuzzy relational equations for a fuzzy relation $R_{\textsf{MA}}$.)

Correctness of Mamdani–Assilian controller

When $\forall j : \Phi(A_j) = A_j \underset{\cdot}{\circ} R_{\mathsf{MA}} = C_j$? (A system of fuzzy relational equations for a fuzzy relation R_{MA}.)

For Mamdani–Assilian controller:

Theorem: $\forall j : \Phi_{\mathsf{MA}}(A_j) \geq C_j$

Proof: $X := A_j$
$\mathcal{D}(X, A_j) = \mathcal{D}(A_j, A_j) = 1$ (due to normality)
$\Phi_{\mathsf{MA}}(A_j)(y) = \max_i(\mathcal{D}(A_j, A_i) \underset{\cdot}{\wedge} C_i(y)) \geq \underbrace{\mathcal{D}(A_j, A_j)}_{1} \underset{\cdot}{\wedge} C_j(y) = C_j(y)$

Correctness of Mamdani–Assilian controller

When $\forall j : \Phi(A_j) = A_j \underset{\cdot}{\circ} R_{\mathsf{MA}} = C_j$? (A system of fuzzy relational equations for a fuzzy relation R_{MA}.)

For Mamdani–Assilian controller:

Theorem: $\forall j : \Phi_{\mathsf{MA}}(A_j) \geq C_j$

Proof: $X := A_j$
$\mathcal{D}(X, A_j) = \mathcal{D}(A_j, A_j) = 1$ (due to normality)
$\Phi_{\mathsf{MA}}(A_j)(y) = \max_i (\mathcal{D}(A_j, A_i) \underset{\cdot}{\wedge} C_i(y)) \geq \underbrace{\mathcal{D}(A_j, A_j)}_{1} \underset{\cdot}{\wedge} C_j(y) = C_j(y)$

Correctness of Mamdani–Assilian controller

Theorem [de Baets 1996, Perfilieva, Tonis 1997]: $\big(\forall j : \Phi_{\mathsf{MA}}(A_j) = C_j\big)$ iff $\big(\forall i\ \forall j : \mathcal{D}(A_i, A_j) \le \mathcal{I}(C_i, C_j)\big)$,
where $\mathcal{I}(C_i, C_j) = \inf\limits_y \big(C_i(y) \underset{\cdot}{\rightarrow} C_j(y)\big)$
(the implication $\underset{\cdot}{\rightarrow}$ has to be the residuum of $\underset{\cdot}{\wedge}$)

Instead of $I(C_i, C_j)$ we may use $\mathcal{E}(C_i, C_j) = \inf\limits_y \big(C_i(y) \underset{\cdot}{\leftrightarrow} C_j(y)\big)$
(**degree of indistinguishability (equality)**),
where $\alpha \underset{\cdot}{\leftrightarrow} \beta = \min(\alpha \underset{\cdot}{\rightarrow} \beta, \beta \underset{\cdot}{\rightarrow} \alpha) = (\alpha \underset{\cdot}{\rightarrow} \beta) \underset{\cdot}{\wedge} (\beta \underset{\cdot}{\rightarrow} \alpha)$

Proof: The negation of the left-hand side is

$$
\begin{aligned}
\exists j\ \exists y : \Phi_{\mathsf{MA}}(A_j)(y) &> C_j(y) \\
\exists j\ \exists y\ \exists x : A_j(x) \underset{\cdot}{\wedge} R_{\mathsf{MA}}(x, y) &> C_j(y) \\
\exists i\ \exists j\ \exists y\ \exists x : A_j(x) \underset{\cdot}{\wedge} A_i(x) \underset{\cdot}{\wedge} C_i(y) &> C_j(y) \\
\exists i\ \exists j\ \exists y\ \exists x : A_j(x) \underset{\cdot}{\wedge} A_i(x) &> C_i(y) \underset{\cdot}{\rightarrow} C_j(y) \\
\exists i\ \exists j : \sup_x \big(A_j(x) \underset{\cdot}{\wedge} A_i(x)\big) &> \inf_y \big(C_i(y) \underset{\cdot}{\rightarrow} C_j(y)\big)
\end{aligned}
$$

which is the negation of the right-hand side.

Correctness of Mamdani–Assilian controller

When $\forall j : \Phi(A_j) = A_j \underset{\cdot}{\circ} R_{\mathsf{MA}} = C_j$? (A system of fuzzy relational equations for a fuzzy relation R_{MA}.)

For Mamdani–Assilian controller:

Theorem: $\forall j : \Phi_{\mathsf{MA}}(A_j) \geq C_j$

Proof: $X := A_j$

$\mathcal{D}(X, A_j) = \mathcal{D}(A_j, A_j) = 1$ (due to normality)

$\Phi_{\mathsf{MA}}(A_j)(y) = \max_i(\mathcal{D}(A_j, A_i) \underset{\cdot}{\wedge} C_i(y)) \geq \underbrace{\mathcal{D}(A_j, A_j)}_{1} \underset{\cdot}{\wedge} C_j(y) = C_j(y)$

Correctness of Mamdani–Assilian controller

When $\forall j : \Phi(A_j) = A_j \circ R_{\mathsf{MA}} = C_j$? (A system of fuzzy relational equations for a fuzzy relation R_{MA}.)

For Mamdani–Assilian controller:

Theorem: $\forall j : \Phi_{\mathsf{MA}}(A_j) \geq C_j$

Proof: $X := A_j$
$\mathcal{D}(X, A_j) = \mathcal{D}(A_j, A_j) = 1$ (due to normality)
$\Phi_{\mathsf{MA}}(A_j)(y) = \max_i (\mathcal{D}(A_j, A_i) \wedge C_i(y)) \geq \underbrace{\mathcal{D}(A_j, A_j)}_{1} \wedge C_j(y) = C_j(y)$

Correctness of Mamdani–Assilian controller

Theorem [de Baets 1996, Perfilieva, Tonis 1997]: $\big(\forall j : \Phi_{\mathsf{MA}}(A_j) = C_j\big)$ iff $\big(\forall i\ \forall j : \mathcal{D}(A_i, A_j) \le \mathcal{I}(C_i, C_j)\big)$,
where $\mathcal{I}(C_i, C_j) = \inf\limits_y \big(C_i(y) \mathbin{\dot\to} C_j(y)\big)$
(the implication $\dot\to$ has to be the residuum of $\dot\wedge$)

Instead of $I(C_i, C_j)$ we may use $\mathcal{E}(C_i, C_j) = \inf\limits_y \big(C_i(y) \mathbin{\dot\leftrightarrow} C_j(y)\big)$
(**degree of indistinguishability (equality)**),
where $\alpha \mathbin{\dot\leftrightarrow} \beta = \min(\alpha \mathbin{\dot\to} \beta, \beta \mathbin{\dot\to} \alpha) = (\alpha \mathbin{\dot\to} \beta) \mathbin{\dot\wedge} (\beta \mathbin{\dot\to} \alpha)$

Proof: The negation of the left-hand side is

$$\begin{aligned}
\exists j\ \exists y : \Phi_{\mathsf{MA}}(A_j)(y) &> C_j(y) \\
\exists j\ \exists y\ \exists x : A_j(x) \mathbin{\dot\wedge} R_{\mathsf{MA}}(x, y) &> C_j(y) \\
\exists i\ \exists j\ \exists y\ \exists x : A_j(x) \mathbin{\dot\wedge} A_i(x) \mathbin{\dot\wedge} C_i(y) &> C_j(y) \\
\exists i\ \exists j\ \exists y\ \exists x : A_j(x) \mathbin{\dot\wedge} A_i(x) &> C_i(y) \mathbin{\dot\to} C_j(y) \\
\exists i\ \exists j : \sup_x \big(A_j(x) \mathbin{\dot\wedge} A_i(x)\big) &> \inf_y \big(C_i(y) \mathbin{\dot\to} C_j(y)\big)
\end{aligned}$$

which is the negation of the right-hand side.

Correctness of Mamdani–Assilian controller

Theorem [de Baets 1996, Perfilieva, Tonis 1997]: $\big(\forall j : \Phi_{\mathsf{MA}}(A_j) = C_j\big)$ iff
$\big(\forall i\ \forall j : \mathcal{D}(A_i, A_j) \leq \mathcal{I}(C_i, C_j)\big)$ $\wedge\ I(C_j, C_i)$
where $\mathcal{I}(C_i, C_j) = \inf_y \big(C_i(y) \underset{\cdot}{\rightarrow} C_j(y)\big)$
(the implication $\underset{\cdot}{\rightarrow}$ has to be the residuum of $\underset{\cdot}{\wedge}$)

Instead of $I(C_i, C_j)$ we may use $\mathcal{E}(C_i, C_j) = \inf_y \big(C_i(y) \underset{\cdot}{\leftrightarrow} C_j(y)\big)$
(**degree of indistinguishability (equality)**),
where $\alpha \underset{\cdot}{\leftrightarrow} \beta = \min(\alpha \underset{\cdot}{\rightarrow} \beta, \beta \underset{\cdot}{\rightarrow} \alpha) = (\alpha \underset{\cdot}{\rightarrow} \beta) \underset{\cdot}{\wedge} (\beta \underset{\cdot}{\rightarrow} \alpha)$

Proof: The negation of the left-hand side is

$$
\begin{aligned}
\exists j\ \exists y : \Phi_{\mathsf{MA}}(A_j)(y) &> C_j(y) \\
\exists j\ \exists y\ \exists x : A_j(x) \underset{\cdot}{\wedge} R_{\mathsf{MA}}(x, y) &> C_j(y) \\
\exists i\ \exists j\ \exists y\ \exists x : A_j(x) \underset{\cdot}{\wedge} A_i(x) \underset{\cdot}{\wedge} C_i(y) &> C_j(y) \\
\exists i\ \exists j\ \exists y\ \exists x : A_j(x) \underset{\cdot}{\wedge} A_i(x) &> C_i(y) \underset{\cdot}{\rightarrow} C_j(y) \\
\exists i\ \exists j : \sup_x \big(A_j(x) \underset{\cdot}{\wedge} A_i(x)\big) &> \inf_y \big(C_i(y) \underset{\cdot}{\rightarrow} C_j(y)\big)
\end{aligned}
$$

which is the negation of the right-hand side.

Correctness of Mamdani–Assilian controller [Moser, Navara 1999]

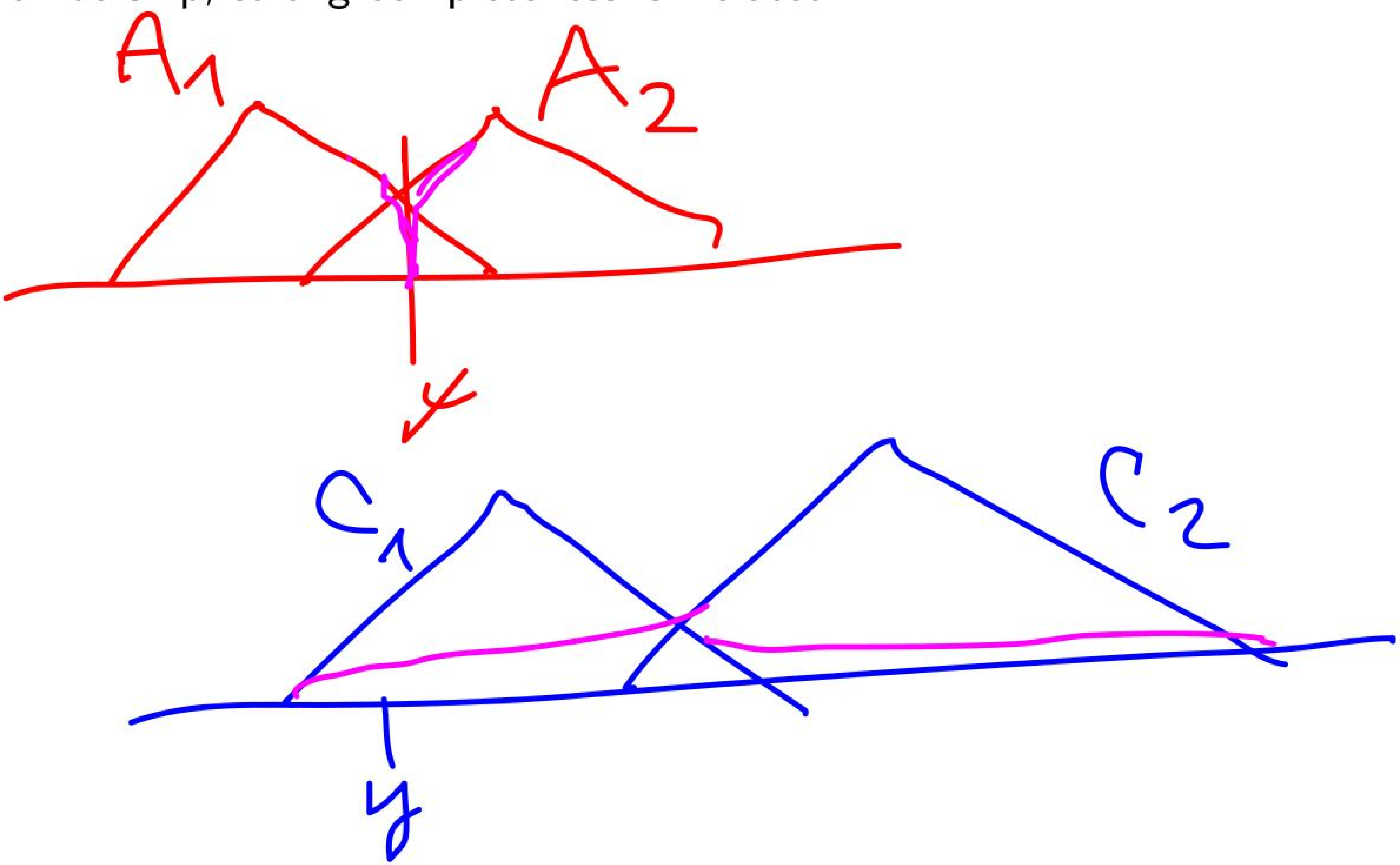
If $\dot{\wedge}$ has no zero divisors (e.g., the minimum or product), then $\mathcal{D}(A_i, A_j) \le \mathcal{E}(C_i, C_j)$ is satisfied in two situations:

- $\mathcal{E}(C_i, C_j) > 0$; then Supp $C_i =$ Supp C_j, which is rather unusual,
- $\mathcal{E}(C_i, C_j) = 0$; then $\mathcal{D}(A_i, A_j) = 0$, Supp $A_i \cap$ Supp $A_j = \emptyset$; for continuous degrees of membership, strong completeness is violated.

Correctness of Mamdani–Assilian controller [Moser, Navara 1999]

If $\wedge$ has no zero divisors (e.g., the minimum or product), then $\mathcal{D}(A_i, A_j) \leq \mathcal{E}(C_i, C_j)$ is satisfied in two situations:

- $\mathcal{E}(C_i, C_j) > 0$; then Supp C_i = Supp C_j, which is rather unusual,
- $\mathcal{E}(C_i, C_j) = 0$; then $\mathcal{D}(A_i, A_j) = 0$, Supp $A_i \cap$ Supp $A_j = \emptyset$; for continuous degrees of membership, strong completeness is violated.



Correctness of Mamdani–Assilian controller [Moser, Navara 1999]

If $\underset{\cdot}{\wedge}$ has no zero divisors (e.g., the minimum or product), then $\mathcal{D}(A_i, A_j) \leq \mathcal{E}(C_i, C_j)$ is satisfied in two situations:

- $\mathcal{E}(C_i, C_j) > 0$; then Supp C_i = Supp C_j, which is rather unusual,
- $\mathcal{E}(C_i, C_j) = 0$; then $\mathcal{D}(A_i, A_j) = 0$, Supp $A_i \cap$ Supp $A_j = \emptyset$; for continuous degrees of membership, strong completeness is violated.

This problem does not occur if $\underset{\cdot}{\wedge}$ has zero divisors (e.g., the Łukasiewicz t-norm)

Correctness of Mamdani–Assilian controller [Moser, Navara 1999]

If $\dot{\wedge}$ has no zero divisors (e.g., the minimum or product), then $\mathcal{D}(A_i, A_j) \leq \mathcal{E}(C_i, C_j)$ is satisfied in two situations:

- $\mathcal{E}(C_i, C_j) > 0$; then $\operatorname{Supp} C_i = \operatorname{Supp} C_j$, which is rather unusual,
- $\mathcal{E}(C_i, C_j) = 0$; then $\mathcal{D}(A_i, A_j) = 0$, $\operatorname{Supp} A_i \cap \operatorname{Supp} A_j = \emptyset$; for continuous degrees of membership, strong completeness is violated.

This problem does not occur if $\dot{\wedge}$ has zero divisors (e.g., the Łukasiewicz t-norm)

However, this choice may easily violate the strong completeness [Moser, Navara 1999]

Correctness of residuum-based controller

Theorem: $\forall j : \Phi_{\mathsf{RES}}(A_j) \le C_j$

Proof: $X := A_j$

$$\begin{aligned}\Phi_{\mathsf{RES}}(A_j)(y) &= \sup_x\big(A_j(x) \mathbin{\dot\wedge} \min_i(A_i(x) \mathbin{\dot\rightarrow} C_i(y))\big)\\ &\le \sup_x\big(A_j(x) \mathbin{\dot\wedge} (A_j(x) \mathbin{\dot\rightarrow} C_j(y))\big) \le C_j(y)\end{aligned}$$

Theorem: If there is a fuzzy relation R such that $\forall j : A_j \mathbin{\dot\circ} R = C_j$, then also R_{RES} satisfies these equalities (and it is the largest solution).

Proof: $\forall j\ \forall x\ \forall y :$

$$\begin{aligned}A_j(x) \mathbin{\dot\wedge} R(x,y) &\le C_j(y)\\ R(x,y) &\le A_j(x) \mathbin{\dot\rightarrow} C_j(y)\\ R(x,y) &\le \min_i\big(A_i(x) \mathbin{\dot\rightarrow} C_i(y)\big) = R_{\mathsf{RES}}(x,y)\end{aligned}$$

$$C_j = A_j \mathbin{\dot\circ} R \le A_j \mathbin{\dot\circ} R_{\mathsf{RES}} \le C_j$$

Correctness of residuum-based controller

Theorem: $\forall j : \Phi_{\mathsf{RES}}(A_j) \leq C_j$

Proof: $X := A_j$

$$\begin{aligned}\Phi_{\mathsf{RES}}(A_j)(y) &= \sup_x \big(A_j(x) \mathbin{\dot\wedge} \min_i (A_i(x) \mathbin{\dot\rightarrow} C_i(y))\big) \\ &\leq \sup_x \big(A_j(x) \mathbin{\dot\wedge} (A_j(x) \mathbin{\dot\rightarrow} C_j(y))\big) \leq C_j(y)\end{aligned}$$

Theorem: If there is a fuzzy relation R such that $\forall j : A_j \mathbin{\dot\circ} R = C_j$, then also R_{RES} satisfies these equalities (and it is the largest solution).

Proof: $\forall j\ \forall x\ \forall y$:

$$\begin{aligned}A_j(x) \mathbin{\dot\wedge} R(x,y) &\leq C_j(y) \\ R(x,y) &\leq A_j(x) \mathbin{\dot\rightarrow} C_j(y) \\ R(x,y) &\leq \min_i \big(A_i(x) \mathbin{\dot\rightarrow} C_i(y)\big) = R_{\mathsf{RES}}(x,y)\end{aligned}$$

$$C_j = A_j \mathbin{\dot\circ} R \leq A_j \mathbin{\dot\circ} R_{\mathsf{RES}} \leq C_j$$

What happens if correctness is violated?

Nothing serious, this is usually accepted and possibly compensated during the tuning

However, it causes a distorted interpretation of (possibly good) control rules

An alternative: CFR (Controller with conditionally firing rules) [Moser, Navara 2002]

1st generalization of Mamdani–Assilian controller:

$\varrho\colon [0,1] \to [0,1]$ ... increasing bijection, e.g., $\varrho(t) = t^r$, $r > 1$, or piecewise linear

Transformation of membership degrees in the input space $\mathcal{X}$

The degrees of overlapping, $\mathcal{D}(A_i \circ \varrho, A_j \circ \varrho)$, may be made arbitrarily small

2nd generalization of Mamdani–Assilian controller:

$\sigma\colon [0,1] \to [c,1]$ ... increasing bijection

$(0 < c < 1)$

Transformation of membership degrees in the output space $\mathcal{Y}$

Output $Y \circ \sigma$ has to be transformed back by $\sigma^{[-1]}$,

so the inference rule is not compositional

(however, the computational complexity remains of the same order)

The degrees of equality, $\mathcal{E}(C_i \circ \sigma, C_j \circ \sigma)$, may be made arbitrarily large

We may satisfy $\mathcal{D}(A_i \circ \varrho, A_j \circ \varrho) \leq \mathcal{E}(C_i \circ \sigma, C_j \circ \sigma)$

Problem 1: $\mathcal{D}(X \circ \varrho, A_i \circ \varrho)$ becomes also small, causing "irrelevant outputs" and violating strong completeness

Problem 2: Correctness and strong completeness are "almost contradictory" for the Mamdani–Assilian controller; sometimes they cannot be satisfied simultaneously for any compositional inference rule

An alternative: CFR (Controller with conditionally firing rules)

So far, we obtained a special case of the generalized FATI inference rule,
where $\pi_i(a,b) = \varrho(a) \dot{\wedge} \sigma(b)$, $\beta = \max$, $\kappa(a,b) = \varrho(a) \dot{\wedge} b$, $Q = \sup \circ \sigma^{[-1]}$

However, we need:

3rd generalization of Mamdani–Assilian controller:
For the **degree of firing** in the inference rule, replace the **degree of overlapping** $\mathcal{D}(X, A_i)$ with the normalized value — **degree of conditional firing**

$$\mathcal{C}_i(X) = \frac{\mathcal{D}(X, A_i)}{\max\limits_j \mathcal{D}(X, A_j)}$$

All the above requirements (in particular correctness and crisp correctness) are satisfied if [Moser, Navara 2002, Navara, Šťastný 2002]:

An alternative: CFR (Controller with conditionally firing rules)

So far, we obtained a special case of the generalized FATI inference rule,
where $\pi_i(a,b) = \varrho(a) \mathbin{\dot\wedge} \sigma(b)$, $\beta = \max$, $\kappa(a,b) = \varrho(a) \mathbin{\dot\wedge} b$, $Q = \sup \circ \sigma^{[-1]}$

However, we need:

3rd generalization of Mamdani–Assilian controller:
For the **degree of firing** in the inference rule, replace the **degree of overlapping** $\mathcal{D}(X, A_i)$ with the normalized value — **degree of conditional firing**

$$\mathcal{C}_i(X) = \frac{\mathcal{D}(X, A_i)}{\max\limits_j \mathcal{D}(X, A_j)}$$

All the above requirements (in particular correctness and crisp correctness) are satisfied if [Moser, Navara 2002, Navara, Šťastný 2002]:

[C1] Each antecedent is normal.

An alternative: CFR (Controller with conditionally firing rules)

So far, we obtained a special case of the generalized FATI inference rule,
where $\pi_i(a,b) = \varrho(a) \wedge \sigma(b), \quad \beta = \max, \quad \kappa(a,b) = \varrho(a) \wedge b, \quad Q = \sup \circ \sigma^{[-1]}$

However, we need:

3rd generalization of Mamdani–Assilian controller:
For the **degree of firing** in the inference rule, replace the **degree of overlapping** $\mathcal{D}(X, A_i)$ with the normalized value — **degree of conditional firing**

$$\mathcal{C}_i(X) = \frac{\mathcal{D}(X, A_i)}{\max\limits_j \mathcal{D}(X, A_j)}$$

All the above requirements (in particular correctness and crisp correctness) are satisfied if [Moser, Navara 2002, Navara, Šťastný 2002]:

[C1] Each antecedent is normal.

[C2] Each point of the input space belongs to the support of some antecedent.

An alternative: CFR (Controller with conditionally firing rules)

So far, we obtained a special case of the generalized FATI inference rule,
where $\pi_i(a,b) = \varrho(a) \dot\wedge \sigma(b), \quad \beta = \max, \quad \kappa(a,b) = \varrho(a) \dot\wedge b, \quad Q = \sup \circ \sigma^{[-1]}$

However, we need:

3rd generalization of Mamdani–Assilian controller:
For the **degree of firing** in the inference rule, replace the **degree of overlapping** $\mathcal{D}(X, A_i)$ with the normalized value — **degree of conditional firing**

$$\mathcal{C}_i(X) = \frac{\mathcal{D}(X, A_i)}{\max_j \mathcal{D}(X, A_j)}$$

All the above requirements (in particular correctness and crisp correctness) are satisfied if [Moser, Navara 2002, Navara, Šťastný 2002]:

[C1] Each antecedent is normal.

[C2] Each point of the input space belongs to the support of some antecedent.

[C3] No consequent is covered by the maximum all other consequents.

An alternative: CFR (Controller with conditionally firing rules)

So far, we obtained a special case of the generalized FATI inference rule,
where $\pi_i(a,b) = \varrho(a) \dot\wedge \sigma(b)$, $\beta = \max$, $\kappa(a,b) = \varrho(a) \dot\wedge b$, $Q = \sup \circ \sigma^{[-1]}$

However, we need:

3rd generalization of Mamdani–Assilian controller:
For the **degree of firing** in the inference rule, replace the **degree of overlapping** $\mathcal{D}(X, A_i)$ with the normalized value — **degree of conditional firing**

$$\mathcal{C}_i(X) = \frac{\mathcal{D}(X, A_i)}{\max\limits_j \mathcal{D}(X, A_j)}$$

All the above requirements (in particular correctness and crisp correctness) are satisfied if [Moser, Navara 2002, Navara, Šťastný 2002]:

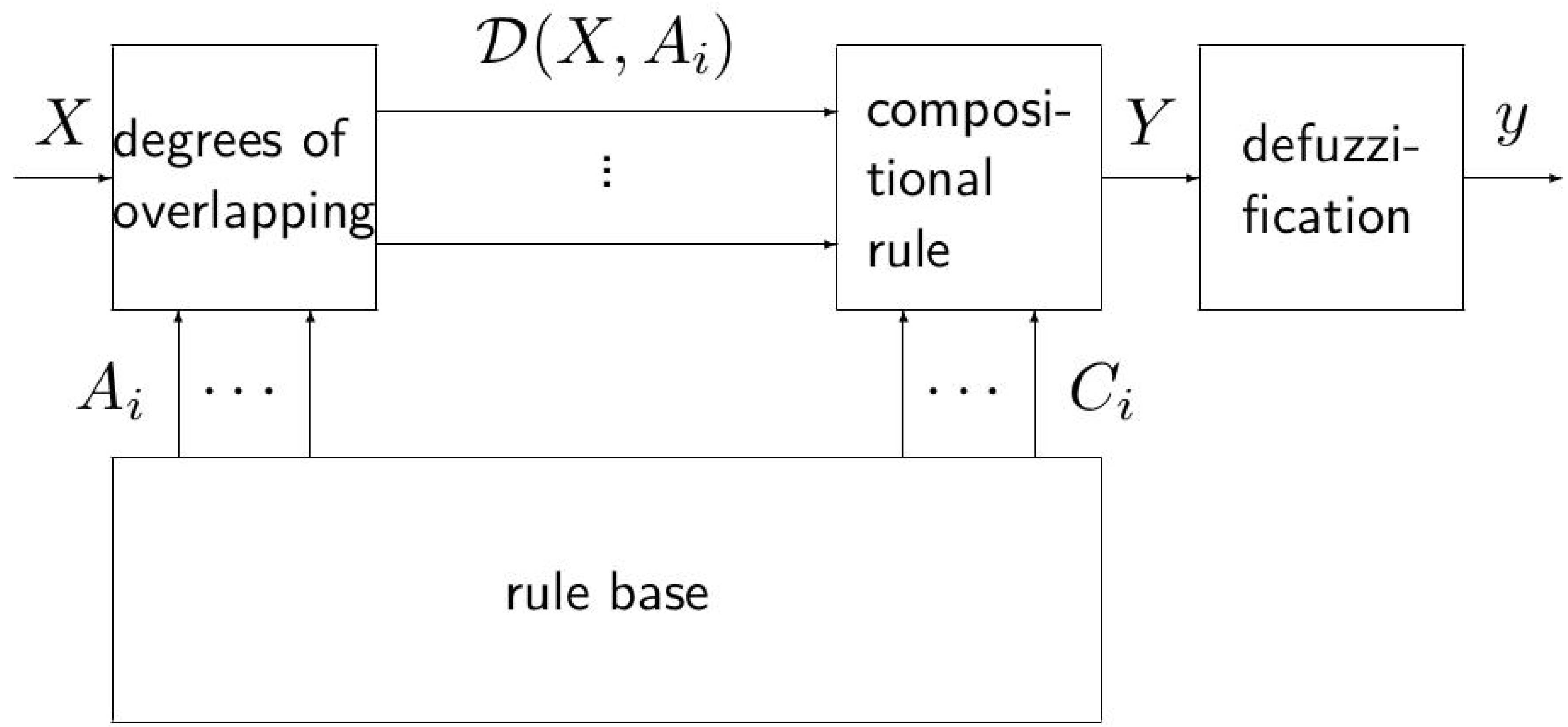
[C1] Each antecedent is normal.

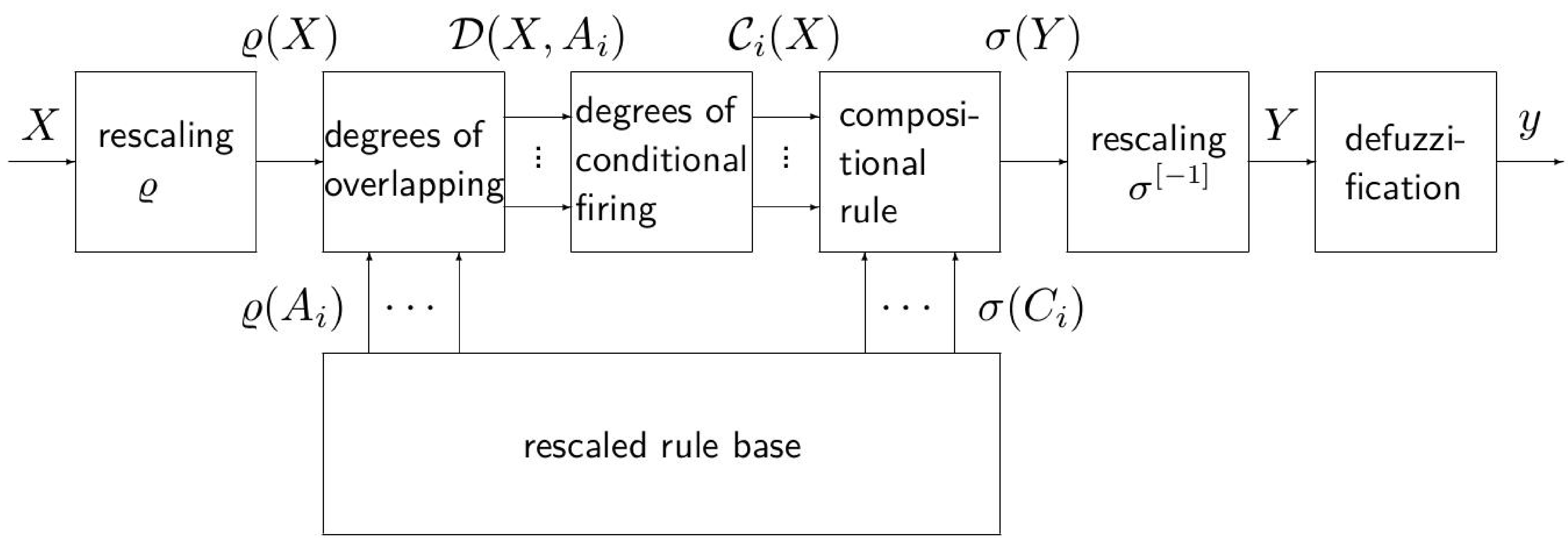
[C2] Each point of the input space belongs to the support of some antecedent.

[C3] No consequent is covered by the maximum all other consequents.

[C4] "Weak disjointness of antecedents": $\exists c < 1 : A_i(x) \dot\wedge A_j(x) < c$ whenever $i \neq j$.

Comparison of Mamdani–Assilian and CFR controller — block diagrams





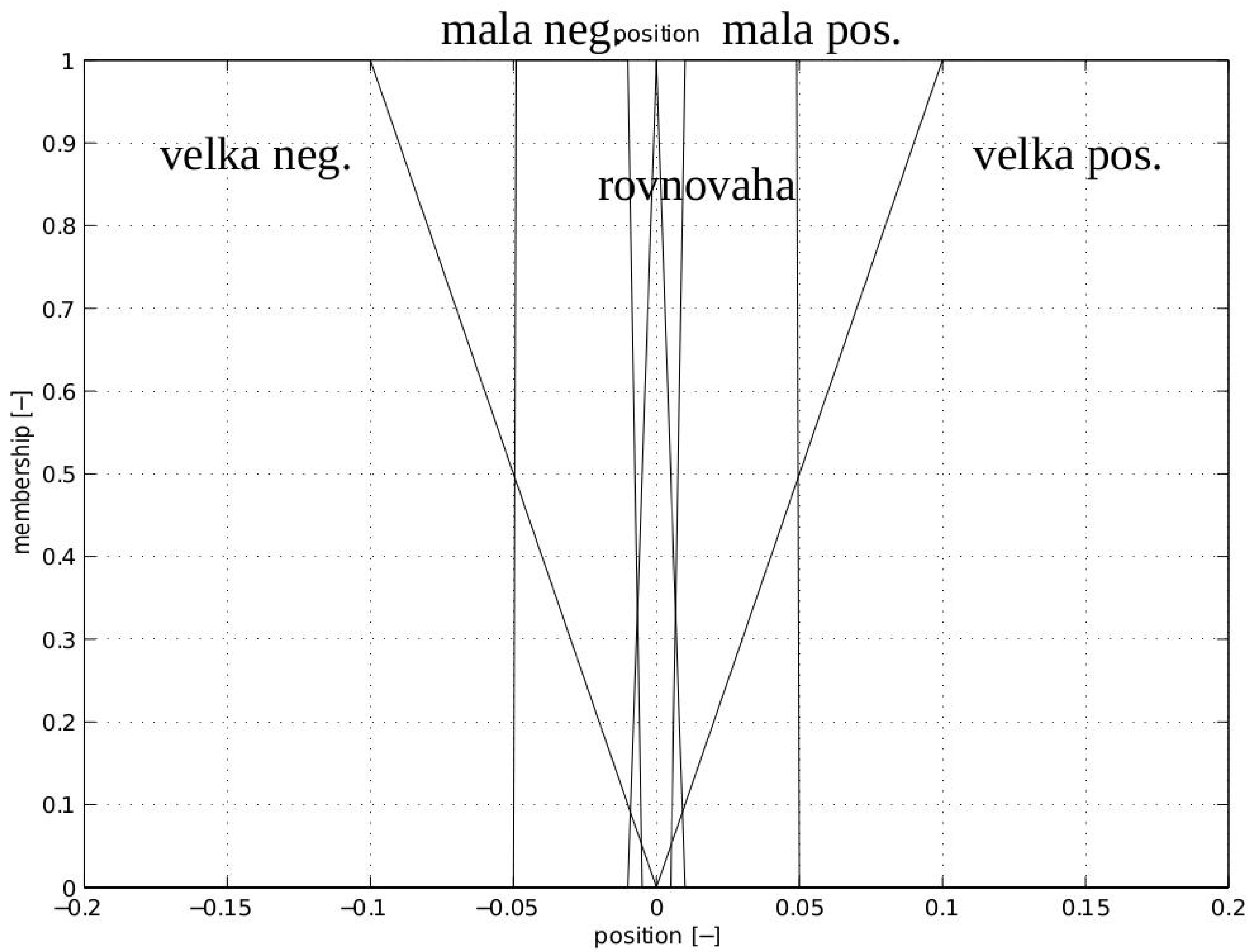
Sample problem: ball on beam (ball on plate)

We want to stabilize a position of a ball by leaning a plate on which it lies

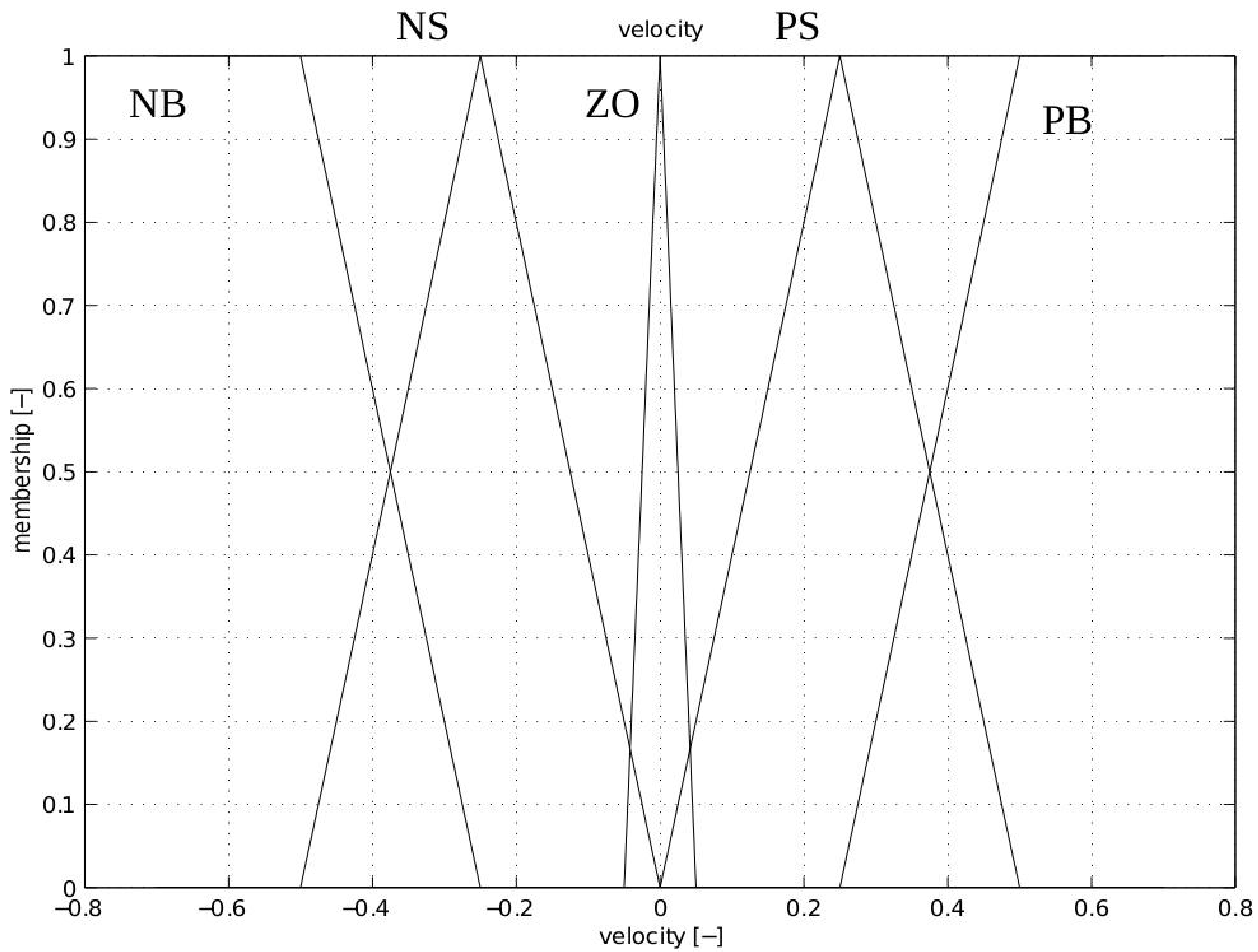
Static friction is considered ($\Rightarrow$ non-linearity)

Solution due to [Šťastný 2001]

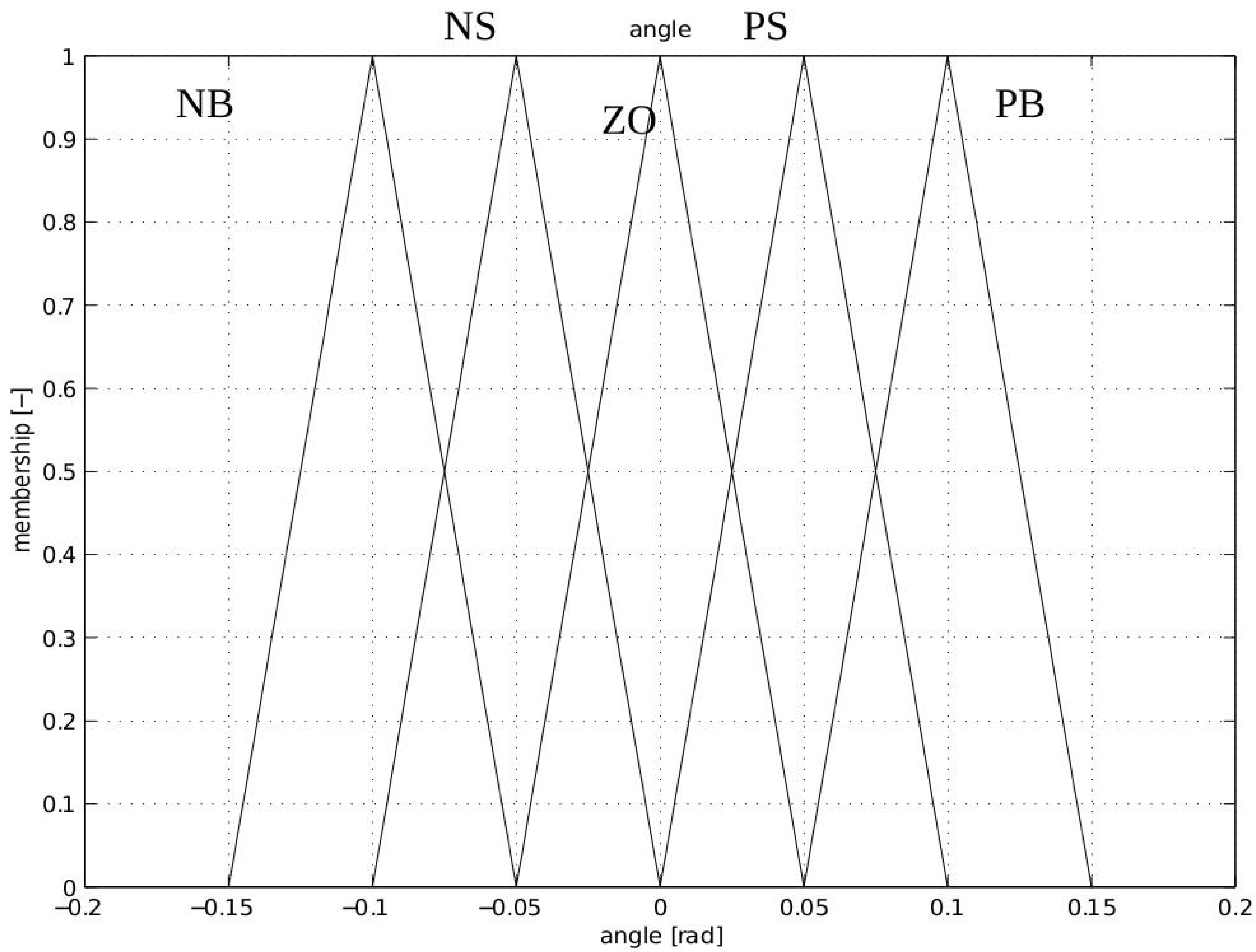
Example: Comparison of Mamdani–Assilian and CFR controller — position (premises)



Example: Comparison of Mamdani–Assilian and CFR controller — velocity (premises)



Example: Comparison of Mamdani–Assilian and CFR controller — angle (consequents)



Example: Comparison of Mamdani–Assilian and CFR controller — rules

Angle:

position / velocity	NB	NS	ZO	PS	PB
NB	PB	PB	PB	PB	PS
NS	PB	PS	PS	PS	ZO
ZO	PB	PB	ZO	NB	NB
PS	ZO	NS	NS	NS	NB
PB	NS	NB	NB	NB	NB

Example: Comparison of Mamdani–Assilian and CFR controller — quality of control

criterion	Mam. controller	CFR controller
maximum overshoot [m] σ	-	-
asymptotic value [m] y_∞	-0.0021	0.0012
number of extremes [-]	-	-
transient time [s]	3.56	3.05
cumulative quadratic error [ms]	0.0552	0.0569

Ball on plate, initial position +0.25, simulation time $5\,s$ — till steady state. Smaller values – better control. $T_{OUT} = 100\,ms$

criterion	Mam. controller	CFR controller
maximum overshoot [m] σ	-	-
asymptotic value [m] y_∞	-0.0052	-0.0006
number of extremes [-]	-	-
transient time [s]	18.06	17.22
cumulative quadratic error [ms]	23.12	22.34

Ball on plate, initial position +2.00, simulation time $20\,s$ — till steady state. Smaller values – better control. $T_{OUT} = 100\,ms$

Example: Comparison of Mamdani–Assilian and CFR controller — quality of control

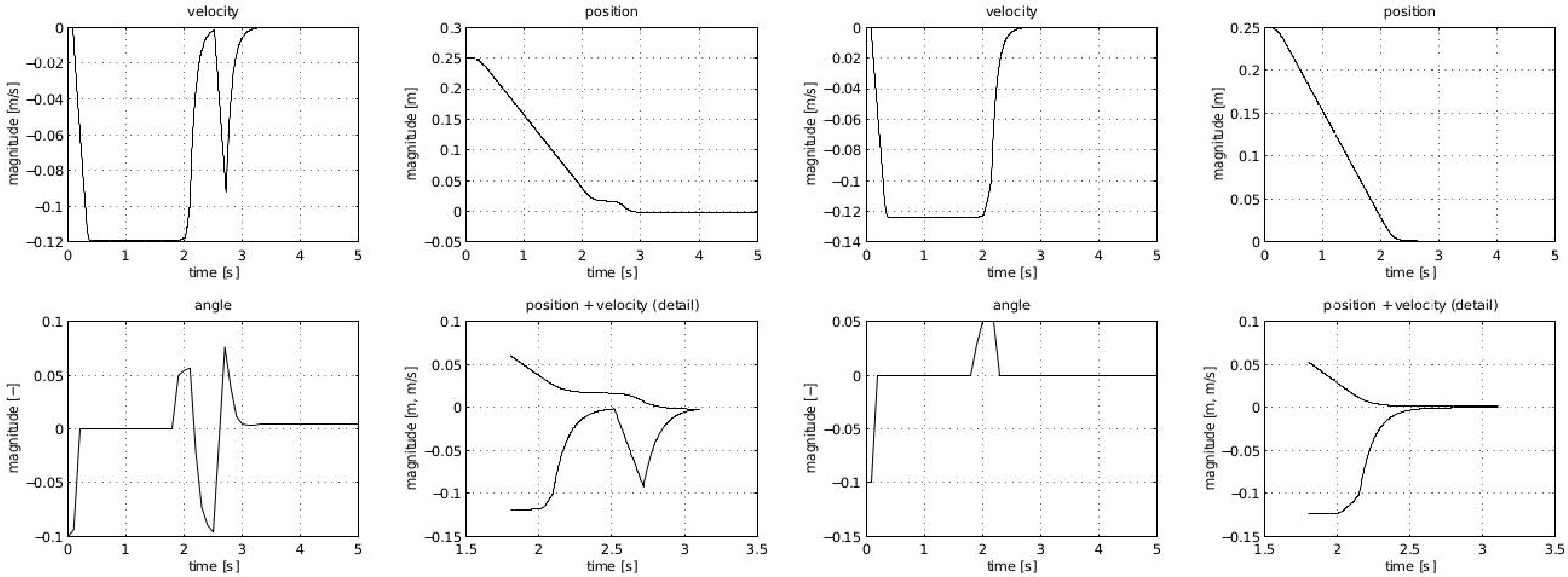
criterion	Mam. controller	CFR controller
maximum overshoot [m] σ	0.35	0.35
asymptotic value [m] y_∞	-0.0032	-0.0041
number of extremes [-]	1	1
transient time [s]	13.49	11.39
cumulative quadratic error [ms]	0.523	0.474

Ball on plate, initial speed $0.5ms^{-1}$, simulation time $15\,s$ — till steady state. Smaller values – better control. $T_{OUT} = 50\,ms$

criterion	Mam. controller	CFR controller
maximum overshoot [m] σ	0.346	0.346
asymptotic value [m] y_∞	0.0051	0.0034
number of extremes [-]	1	1
transient time [s]	14.8	11.1
cumulative quadratic error [ms]	0.583	0.441

Ball on plate, initial speed $0.5ms^{-1}$, simulation time $15\,s$ — till steady state. Smaller values – better control. $T_{OUT} = 5\,ms$

Example: Comparison of Mamdani–Assilian and CFR controller — outputs



Typical outputs of Mamdani–Assilian controller (left) and CFR controller (right)

Problems of implementation of CFR controller

Software implementation: only three new blocks requiring a few lines of source code
The computational complexity slightly increases, but its order remains unchanged

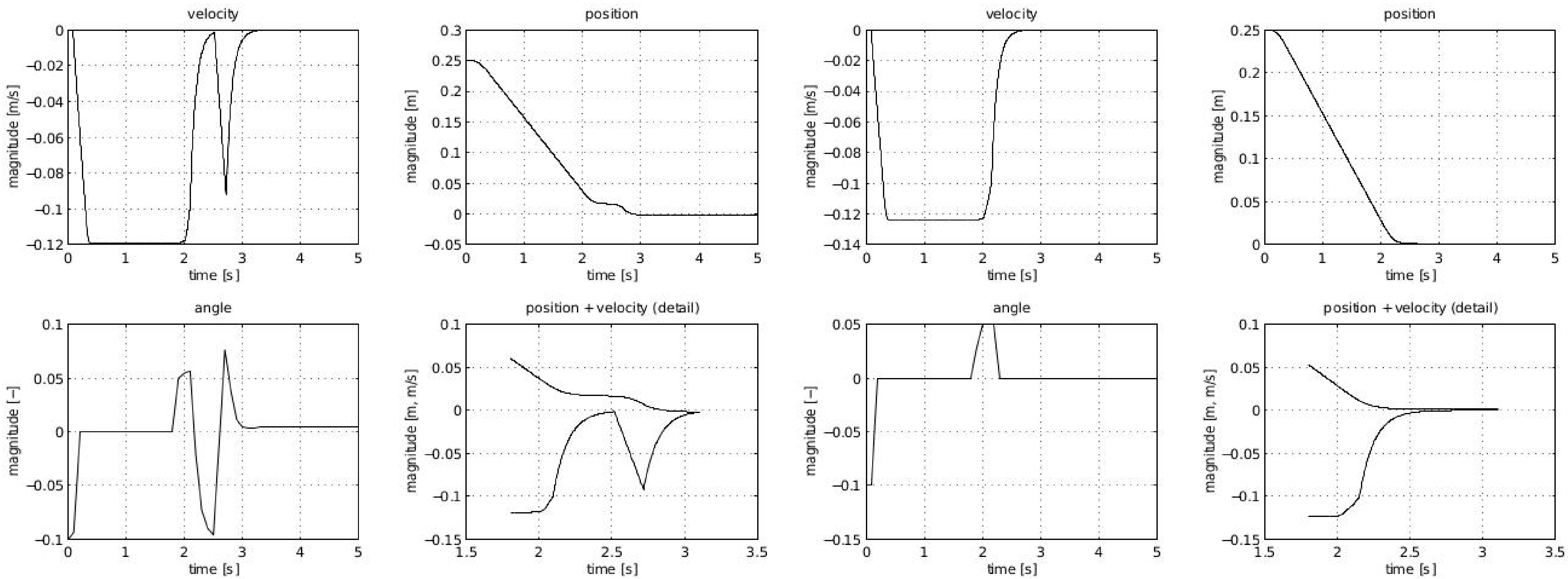
Hardware implementation: Requires to add an additional block inside the current structure, thus a totally new design of an integrated circuit - **expensive!**

Looking for a possibility to achieve the same control action using current fuzzy hardware and a **modified rule base**, we have found [Amato, Di Nola, Navara 2003]:

1. it is not possible to substitute the CFR controller in its full generality, but
2. this is possible for crisp input variables

This case is still of much importance, because it covers most of applications;
in fact, current fuzzy hardware works only with crisp inputs

Example: Comparison of Mamdani–Assilian and CFR controller — outputs



Typical outputs of Mamdani–Assilian controller (left) and CFR controller (right)

Problems of implementation of CFR controller

Software implementation: only three new blocks requiring a few lines of source code
The computational complexity slightly increases, but its order remains unchanged

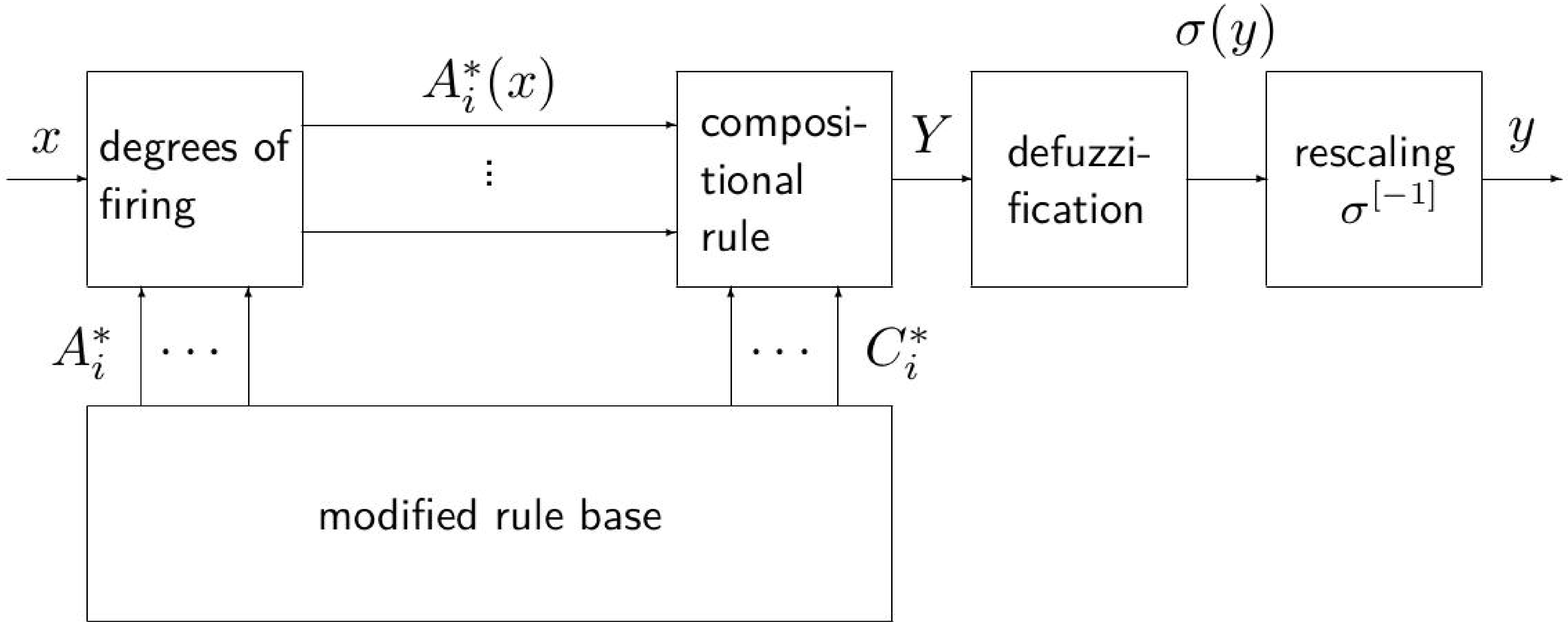
Hardware implementation: Requires to add an additional block inside the current structure, thus a totally new design of an integrated circuit - **expensive!**

Looking for a possibility to achieve the same control action using current fuzzy hardware and a **modified rule base**, we have found [Amato, Di Nola, Navara 2003]:

1. it is not possible to substitute the CFR controller in its full generality, but
2. this is possible for crisp input variables

This case is still of much importance, because it covers most of applications;
in fact, current fuzzy hardware works only with crisp inputs

Hardware implementation of CFR controller



Conclusion

- We formulated well motivated axioms for fuzzy controllers (approximators). They cannot be satisfied by any controller using the classical compositional rule of inference (including the Mamdani–Assilian controller). Our generalized controller satisfies them under very general conditions.
- Practical experiments show that our controller allows to achieve better results with the same rule database.
- The computational efficiency is basically the same as that of the Mamdani–Assilian controller.
- New results allow to transform the rule base (automatically) so that the current fuzzy hardware could be used to implementation of our controller, although its performance could not be achieved by the original Mamdani–Assilian controller.

Initial rule base

Can be obtained by

- asking an expert

Initial rule base

Can be obtained by

- asking an expert
- observing him/her at work

Initial rule base

Can be obtained by

- asking an expert
- observing him/her at work
- combination with analysis of a model (if available)

Initial rule base

Can be obtained by

- asking an expert
- observing him/her at work
- combination with analysis of a model (if available)
- a template for a similar problem

Initial rule base

Can be obtained by

- asking an expert
- observing him/her at work
- combination with analysis of a model (if available)
- a template for a similar problem

Automatic derivation of rules can be made by clustering methods in the space $\mathcal{X} \times \mathcal{Y}$
The clusters are approximated by cylindrical extensions of antecedents and consequents

Tuning

In the phase of tuning, we may

- modify membership functions of antecedents and consequents

Tuning

In the phase of tuning, we may

- modify membership functions of antecedents and consequents
- add new rules

Tuning

In the phase of tuning, we may

- modify membership functions of antecedents and consequents
- add new rules
- delete irrelevant rules or join them with similar ones

Tuning

In the phase of tuning, we may

- modify membership functions of antecedents and consequents
- add new rules
- delete irrelevant rules or join them with similar ones

by

- experimenting with the controller

Tuning

In the phase of tuning, we may

- modify membership functions of antecedents and consequents
- add new rules
- delete irrelevant rules or join them with similar ones

by

- experimenting with the controller
- observing a human controlling the system (interpretability is needed)

Tuning

In the phase of tuning, we may

- modify membership functions of antecedents and consequents
- add new rules
- delete irrelevant rules or join them with similar ones

by

- experimenting with the controller
- observing a human controlling the system (interpretability is needed)

using

- neural networks,

Tuning

In the phase of tuning, we may

- modify membership functions of antecedents and consequents
- add new rules
- delete irrelevant rules or join them with similar ones

by

- experimenting with the controller
- observing a human controlling the system (interpretability is needed)

using

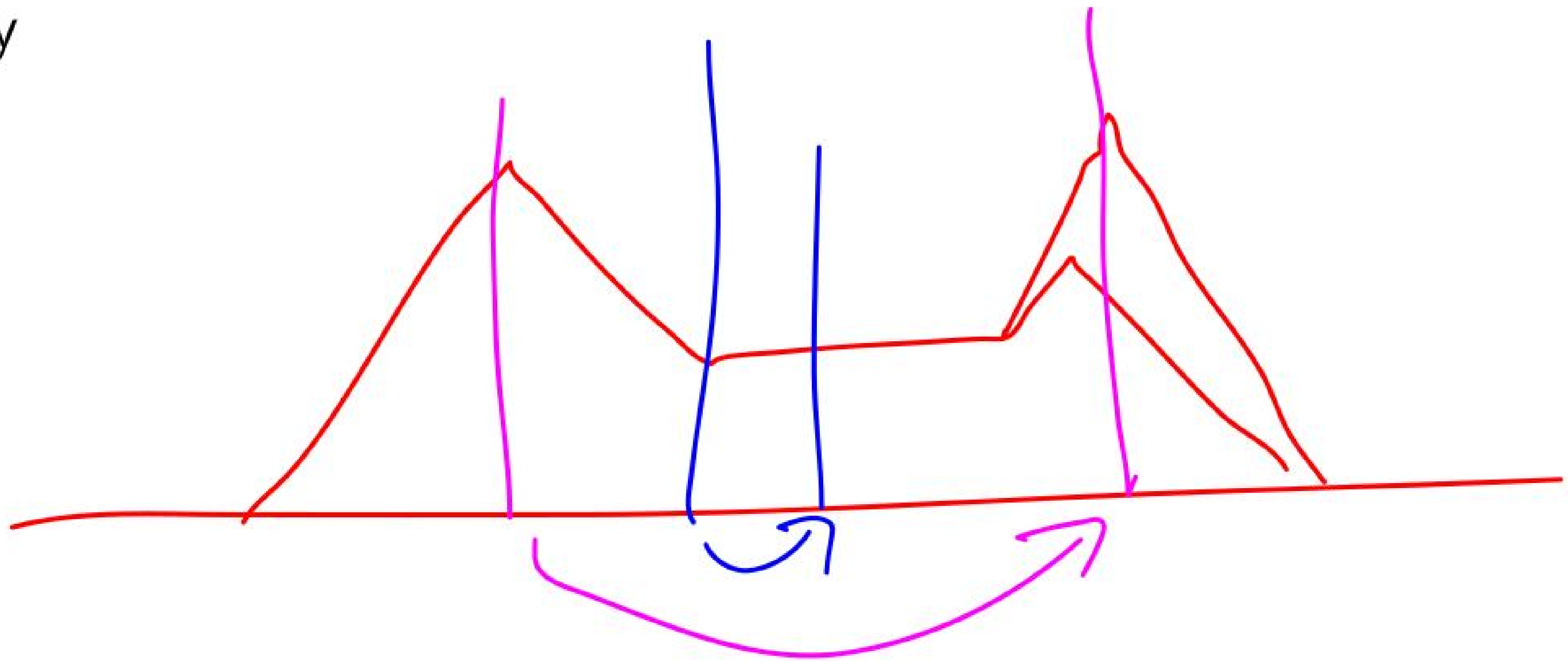
- neural networks,
- genetic algorithms, etc.

Requirements on defuzzification

- Continuity

Requirements on defuzzification

- Continuity



Requirements on defuzzification

- Continuity
- Disambiguity

Requirements on defuzzification

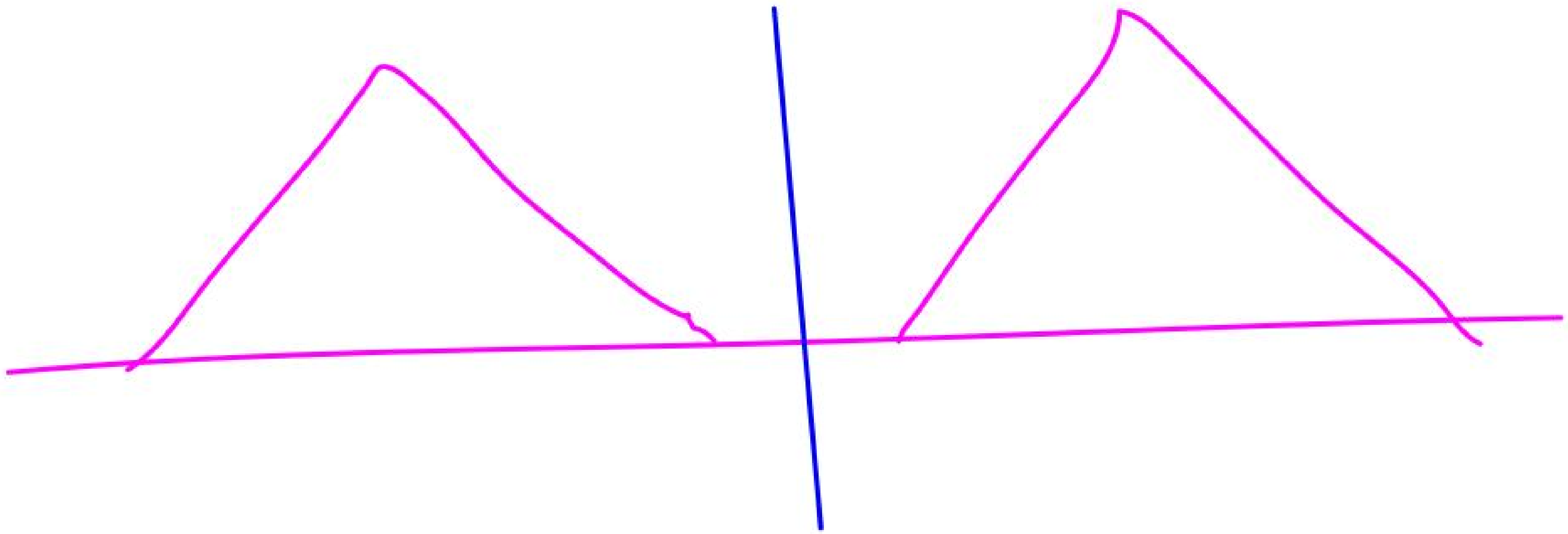
- Continuity
- Disambiguity
- Small computational complexity

Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)

Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)

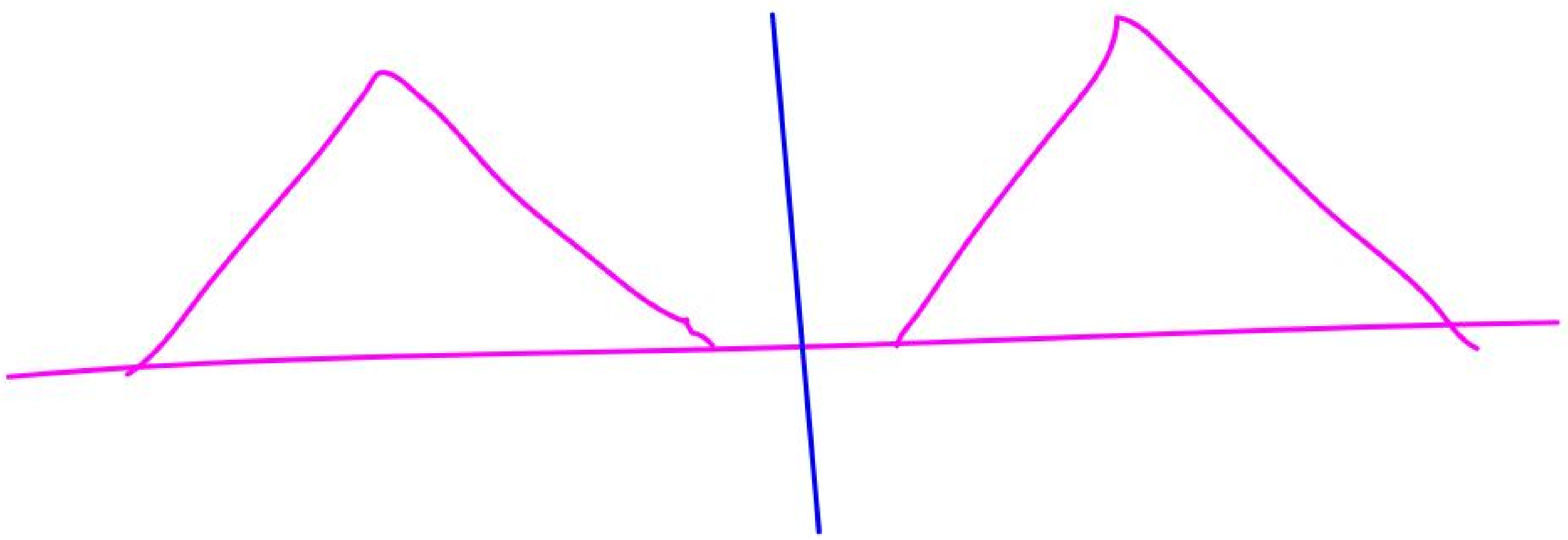


Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)
- Weight counting? (When several firing rules have the same consequent, should we sum them up?)

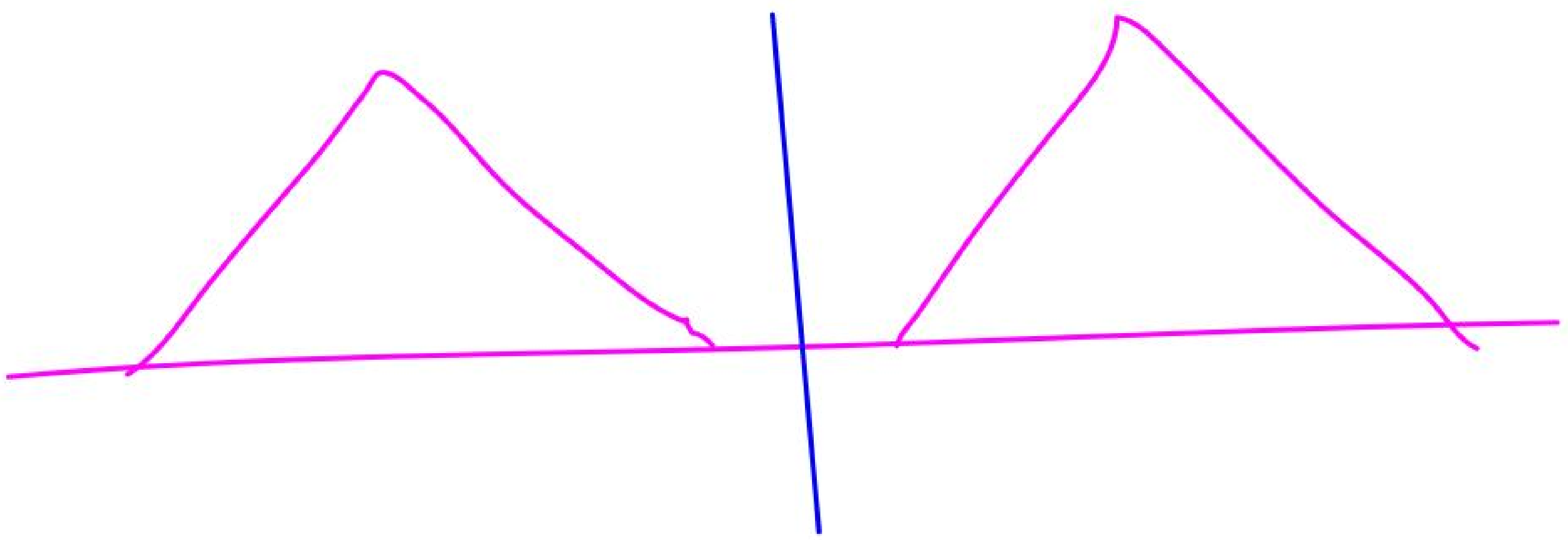
Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)



Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)

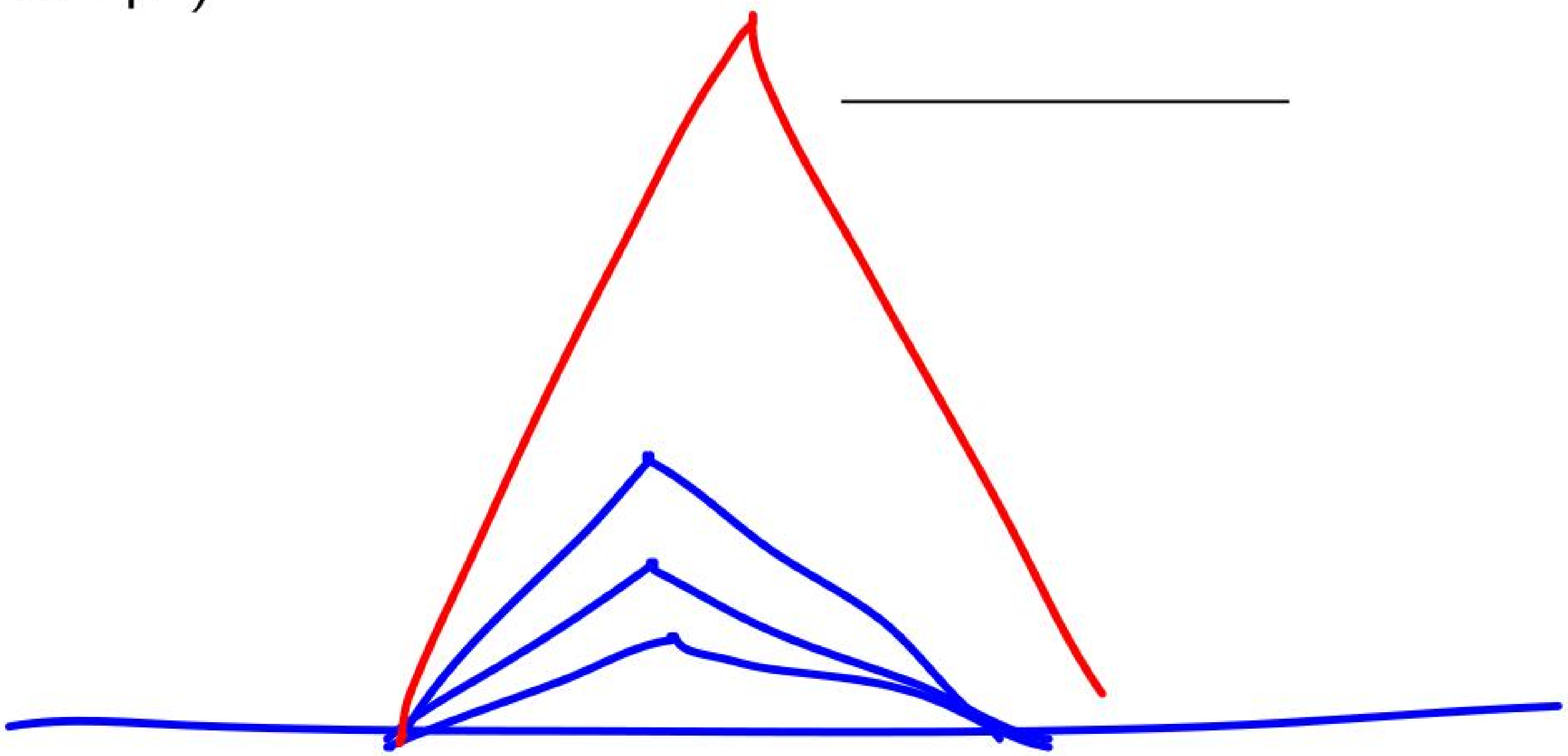


Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)
- Weight counting? (When several firing rules have the same consequent, should we sum them up?)

Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)
- Weight counting? (When several firing rules have the same consequent, should we sum them up?)

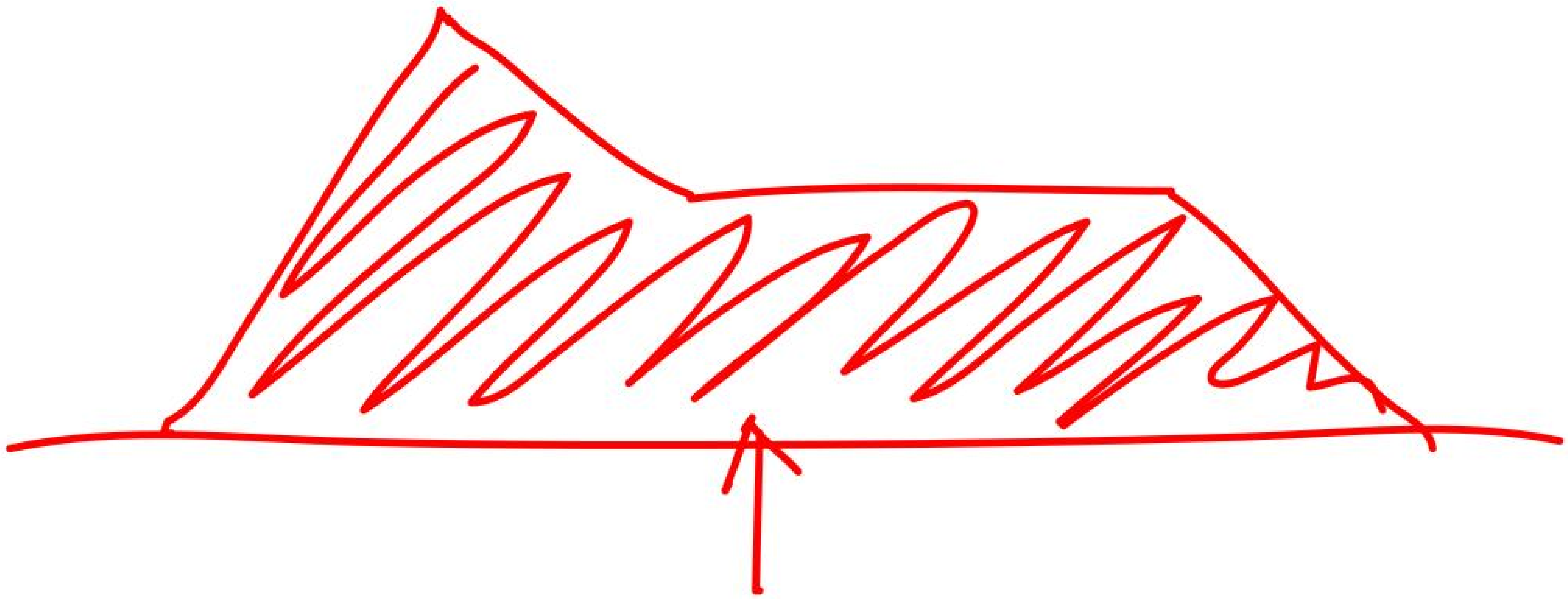


Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
  - Continuity: excellent
  - Disambiguity: none
  - Computational complexity: high
  - Plausibility: doubtful! (it may choose a wrong value between two peeks)

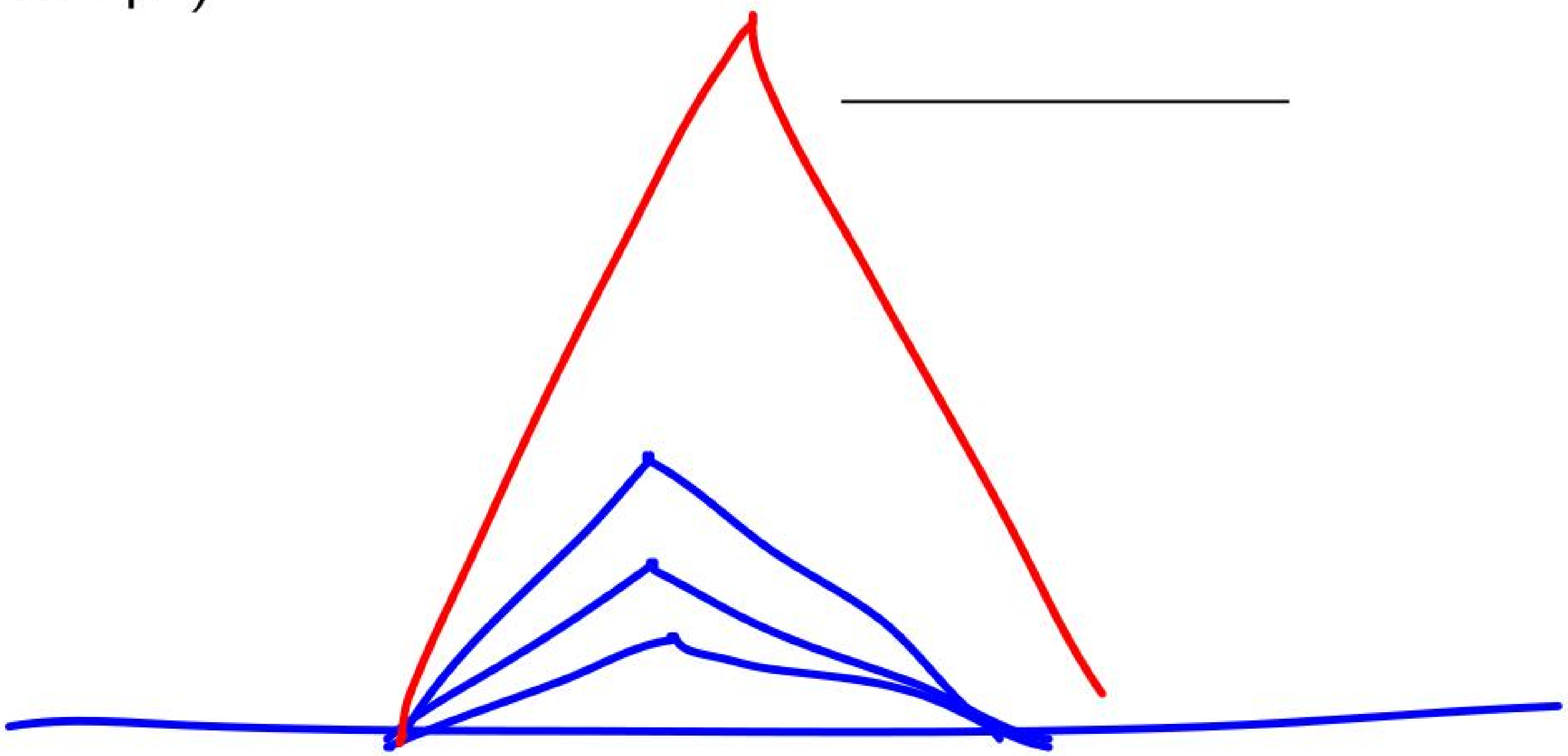
Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
  - Continuity: excellent
  - Disambiguity: none
  - Computational complexity: high
  - Plausibility: doubtful! (it may choose a wrong value between two peeks)



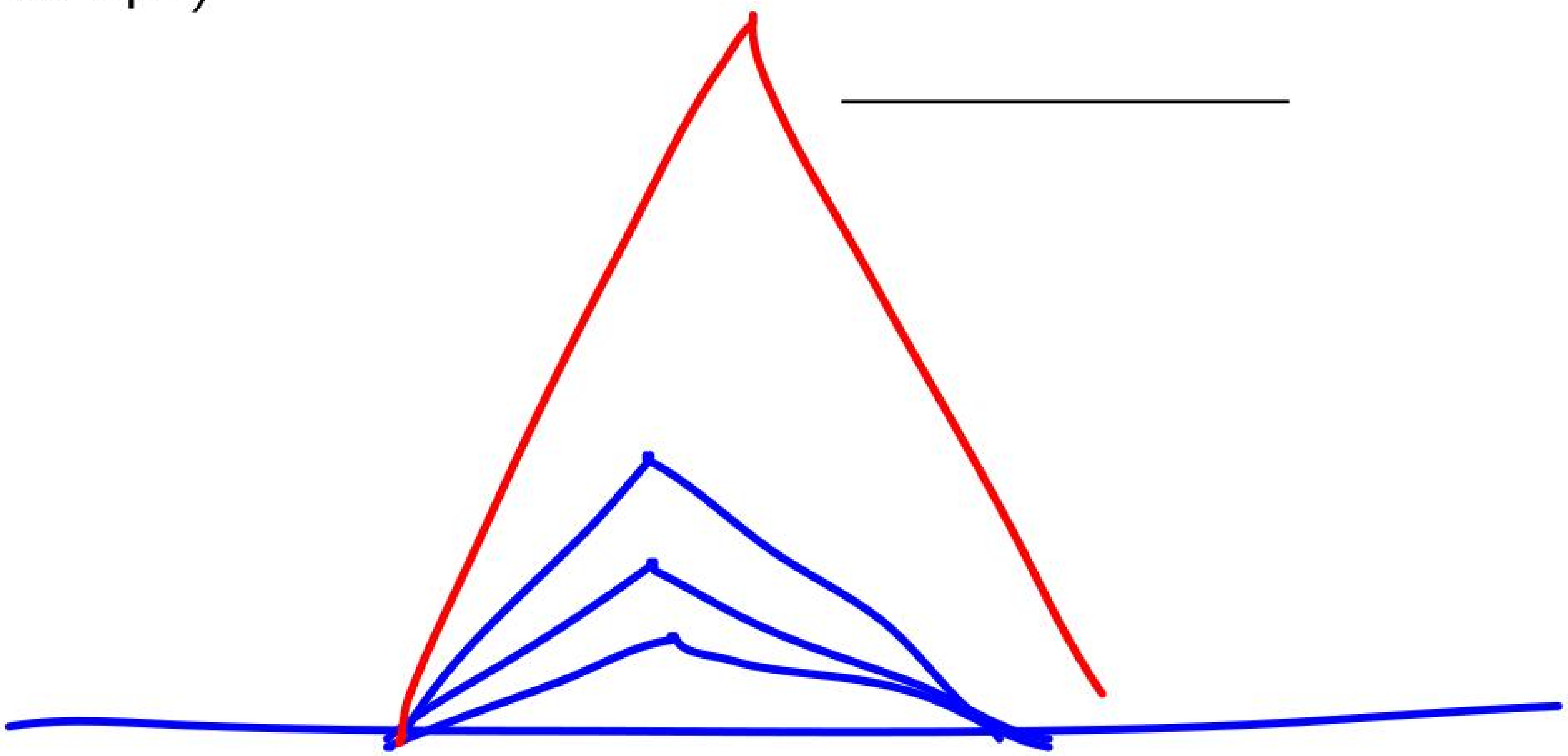
Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)
- Weight counting? (When several firing rules have the same consequent, should we sum them up?)



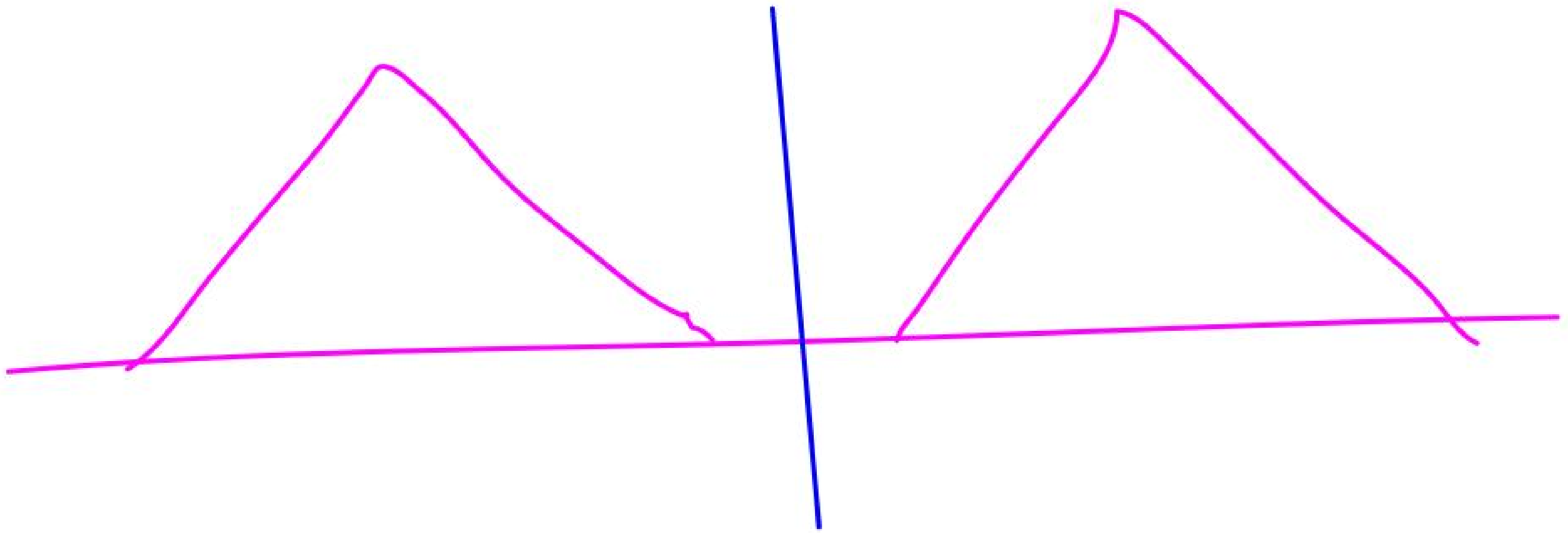
Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)
- Weight counting? (When several firing rules have the same consequent, should we sum them up?)



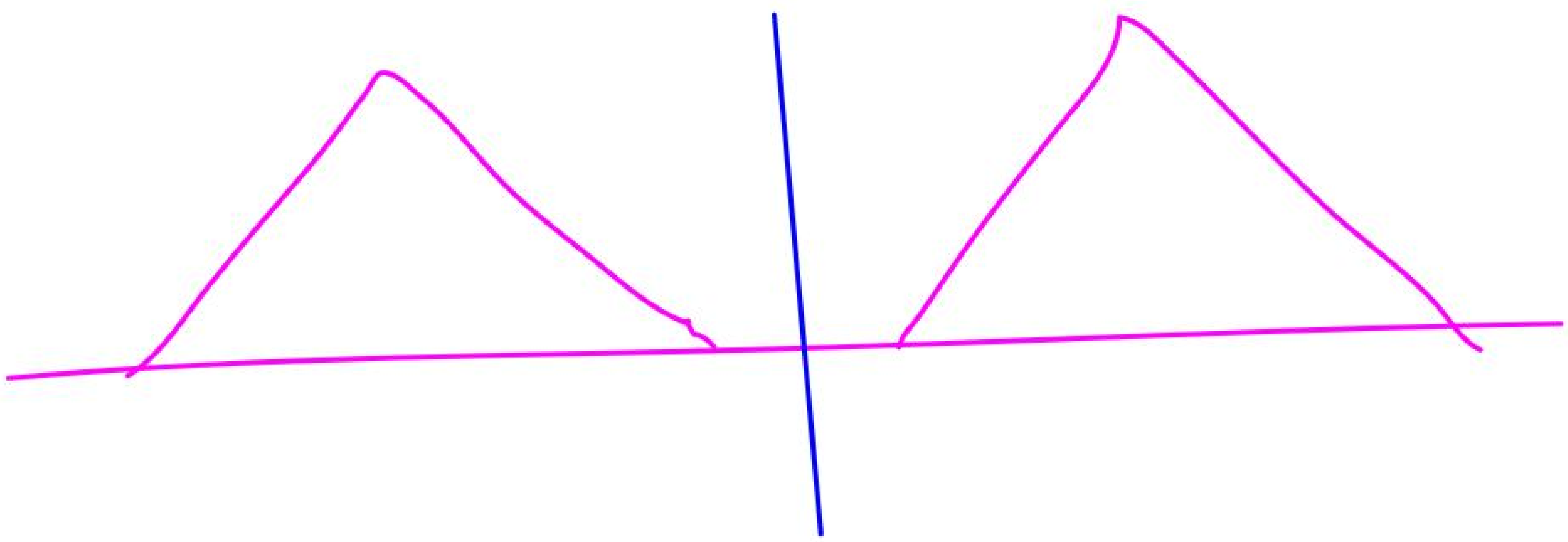
Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)



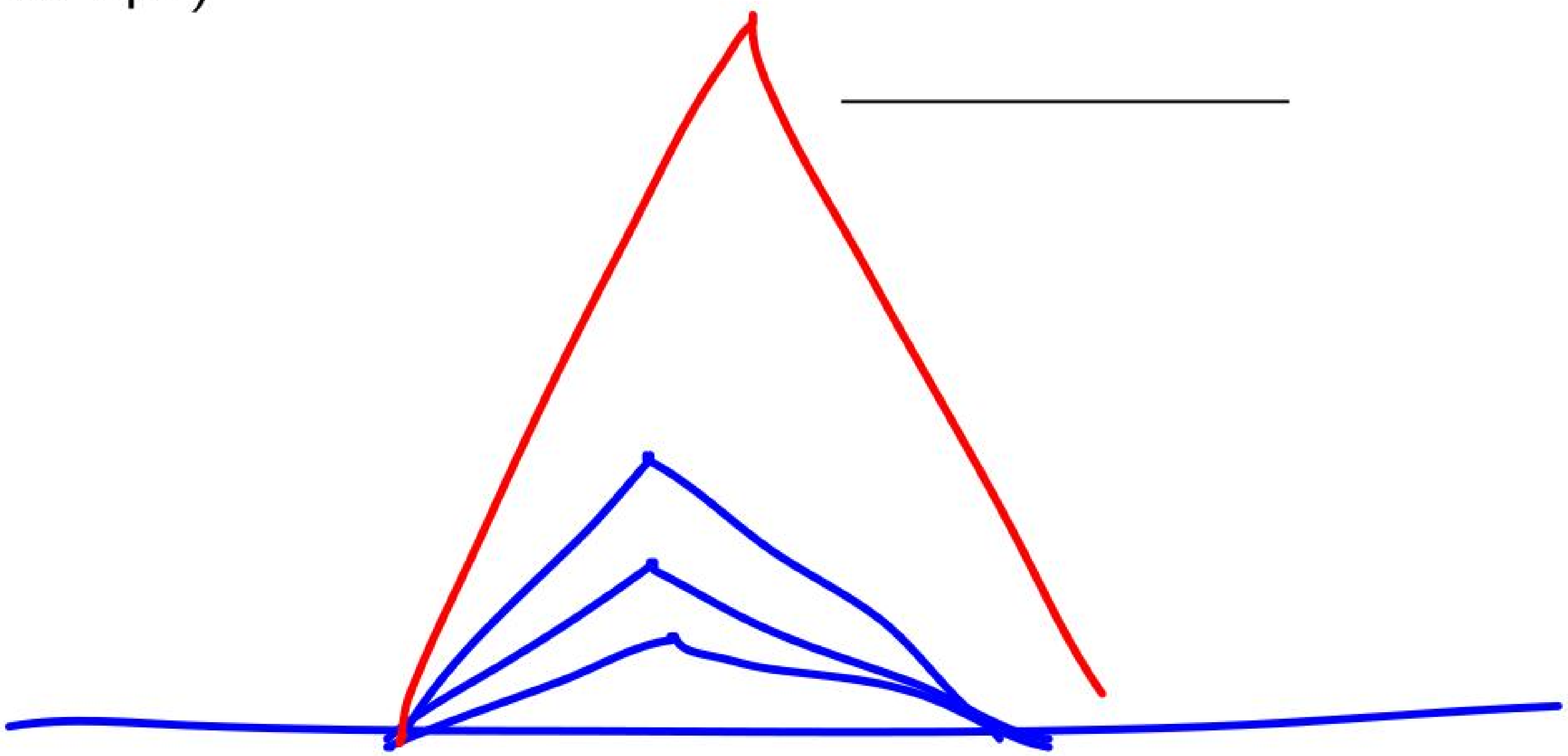
Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)



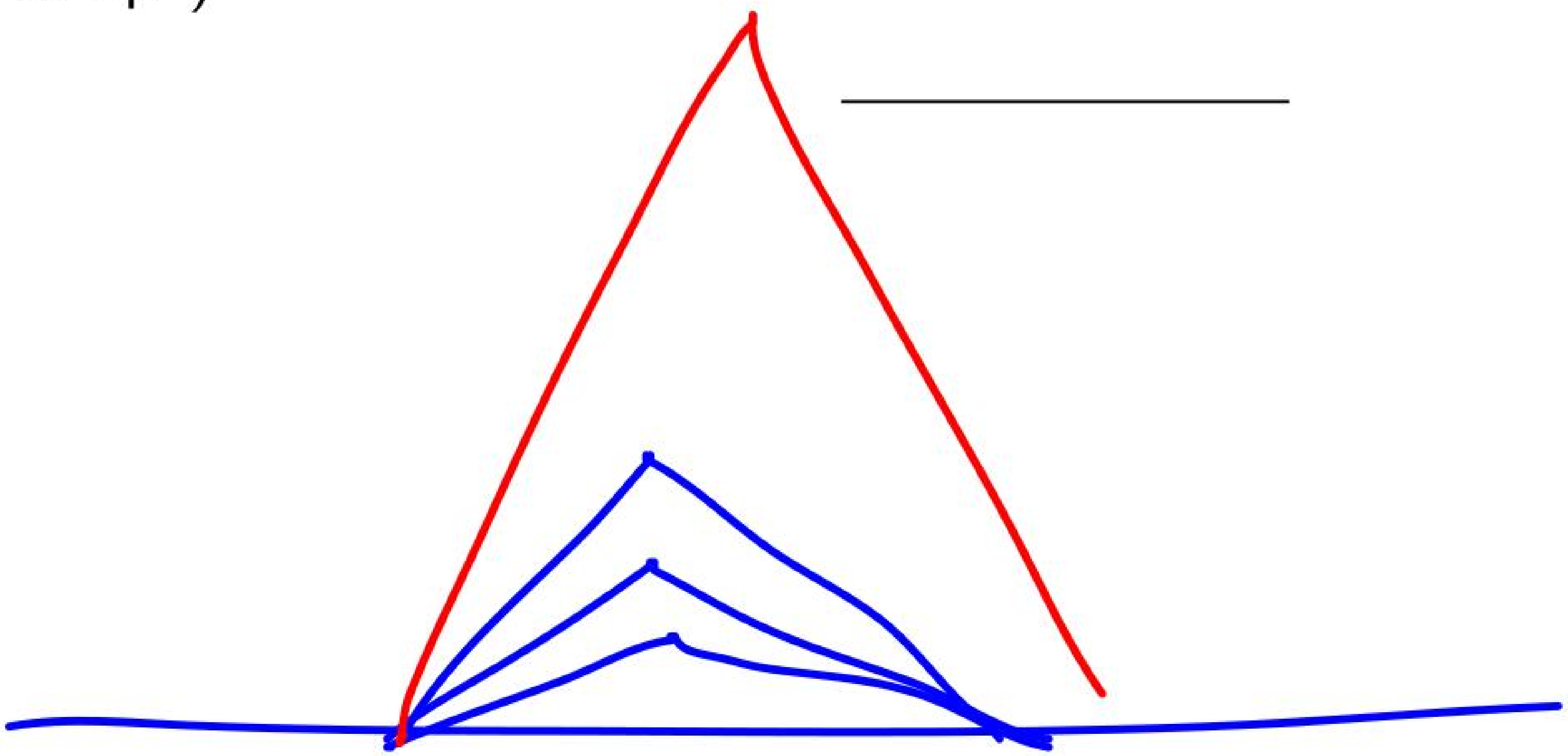
Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)
- Weight counting? (When several firing rules have the same consequent, should we sum them up?)



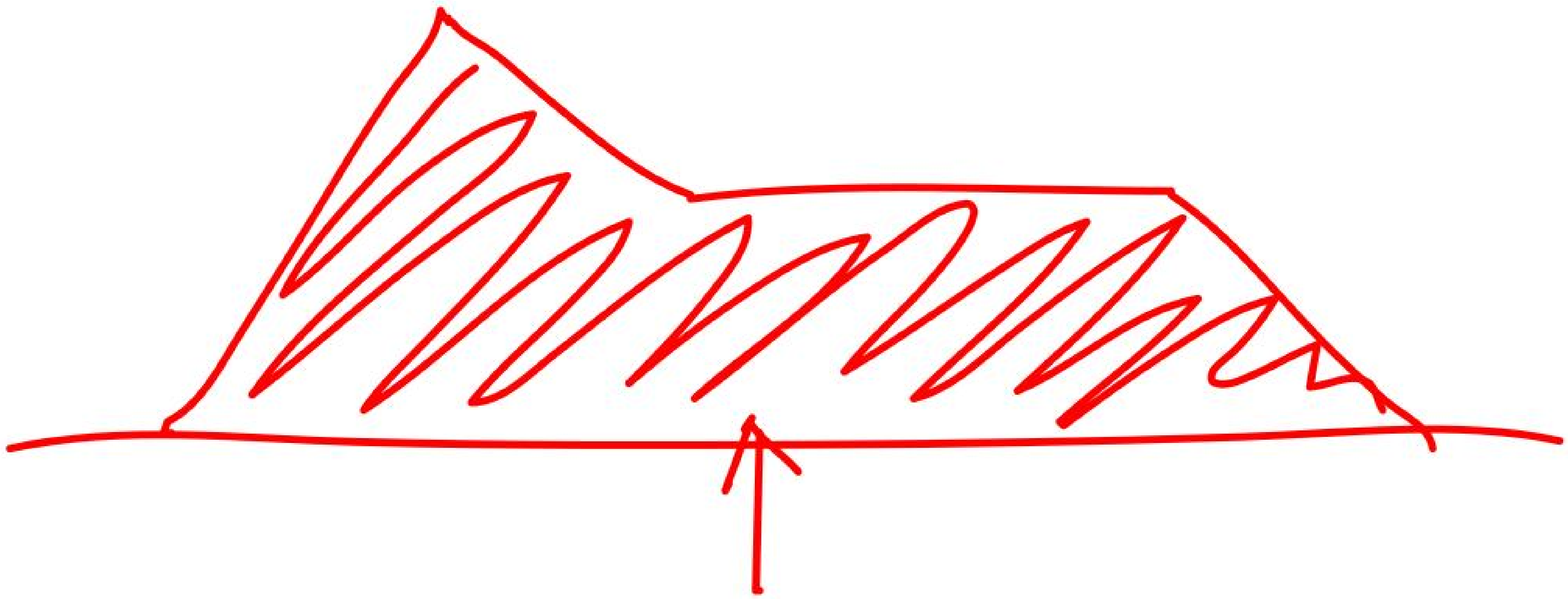
Requirements on defuzzification

- Continuity
- Disambiguity
- Small computational complexity
- Plausibility (the resulting value should be approximately in the middle of the support and have a high degree of membership)
- Weight counting? (When several firing rules have the same consequent, should we sum them up?)



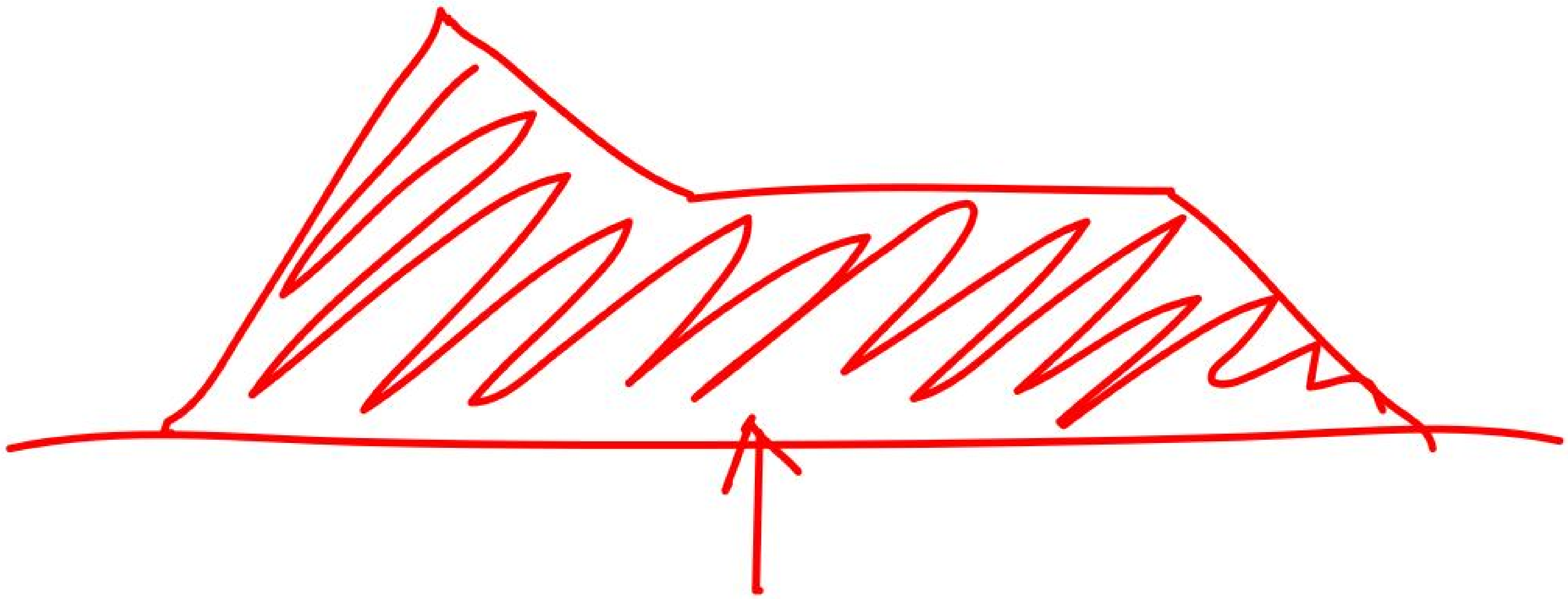
Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
  - Continuity: excellent
  - Disambiguity: none
  - Computational complexity: high
  - Plausibility: doubtful! (it may choose a wrong value between two peeks)



Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
  - Continuity: excellent
  - Disambiguity: none
  - Computational complexity: high
  - Plausibility: doubtful! (it may choose a wrong value between two peeks)

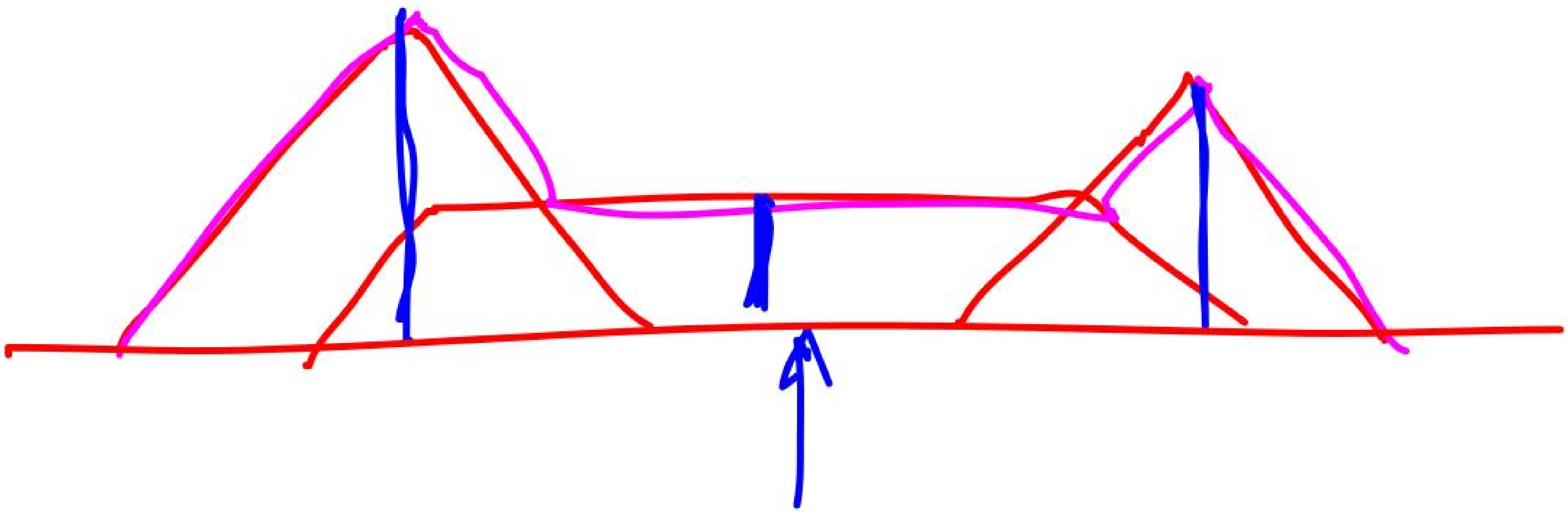


Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
  - Continuity: excellent
  - Disambiguity: none
  - Computational complexity: moderate (centroids corresponding to separate rules may sometimes be computed in advance)
  - Plausibility: doubtful! (it may choose a wrong value between two peeks)

Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
  - Continuity: excellent
  - Disambiguity: none
  - Computational complexity: moderate (centroids corresponding to separate rules may sometimes be computed in advance)
  - Plausibility: doubtful! (it may choose a wrong value between two peeks)

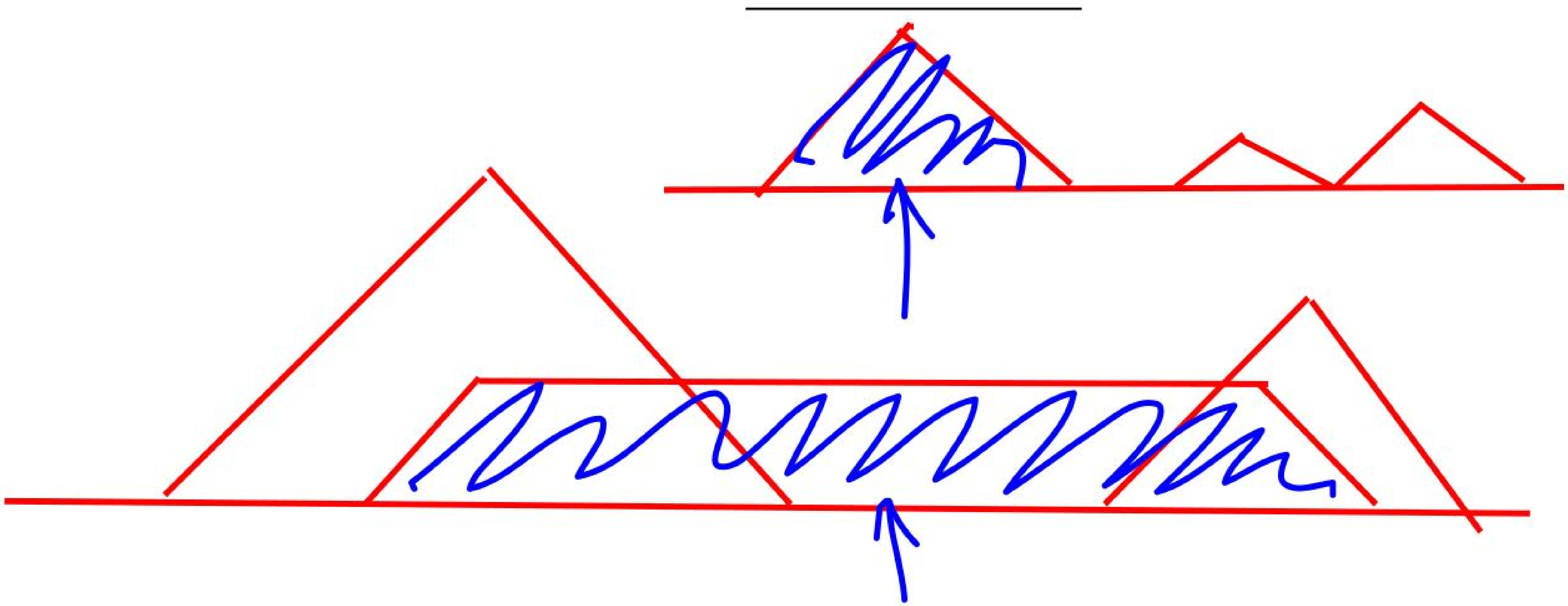


Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
  - Continuity: sometimes violated
  - Disambiguity: sometimes violated
  - Computational complexity: moderate
  - Plausibility: reasonable

Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
  - Continuity: sometimes violated
  - Disambiguity: sometimes violated
  - Computational complexity: moderate
  - Plausibility: reasonable



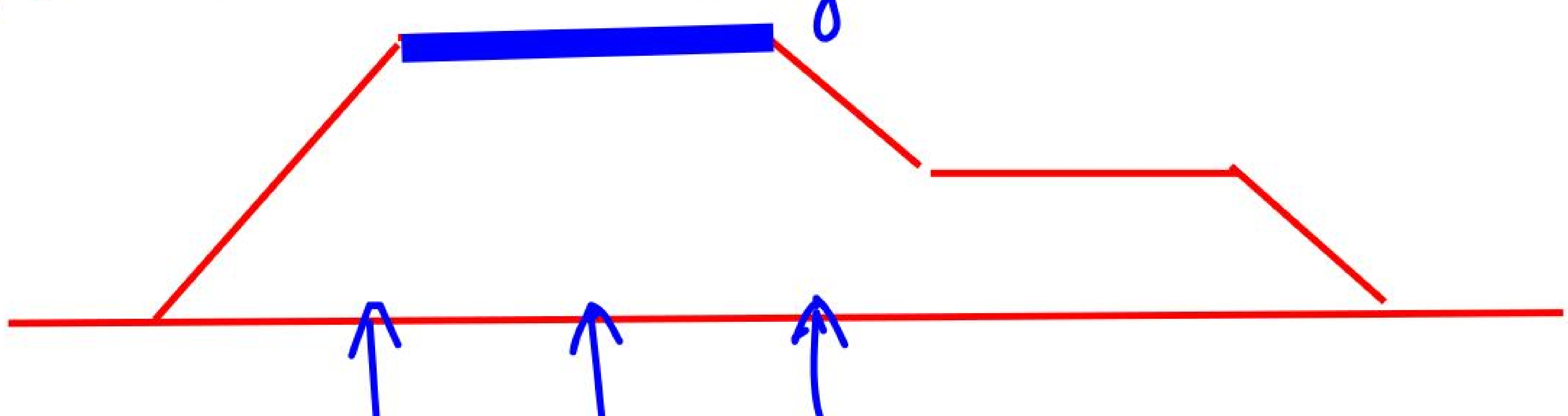
Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
- First/last of maxima
  - Continuity: bad!
  - Disambiguity: only due to an additional criterion
  - Computational complexity: low
  - Plausibility: reasonable

Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
- First/last of maxima
  - Continuity: bad!
  - Disambiguity: only due to an additional criterion
  - Computational complexity: low
  - Plausibility: reasonable

$$y_* = \begin{smallmatrix}\min \\ \max\end{smallmatrix} \arg\max_{y} \Phi(X)$$

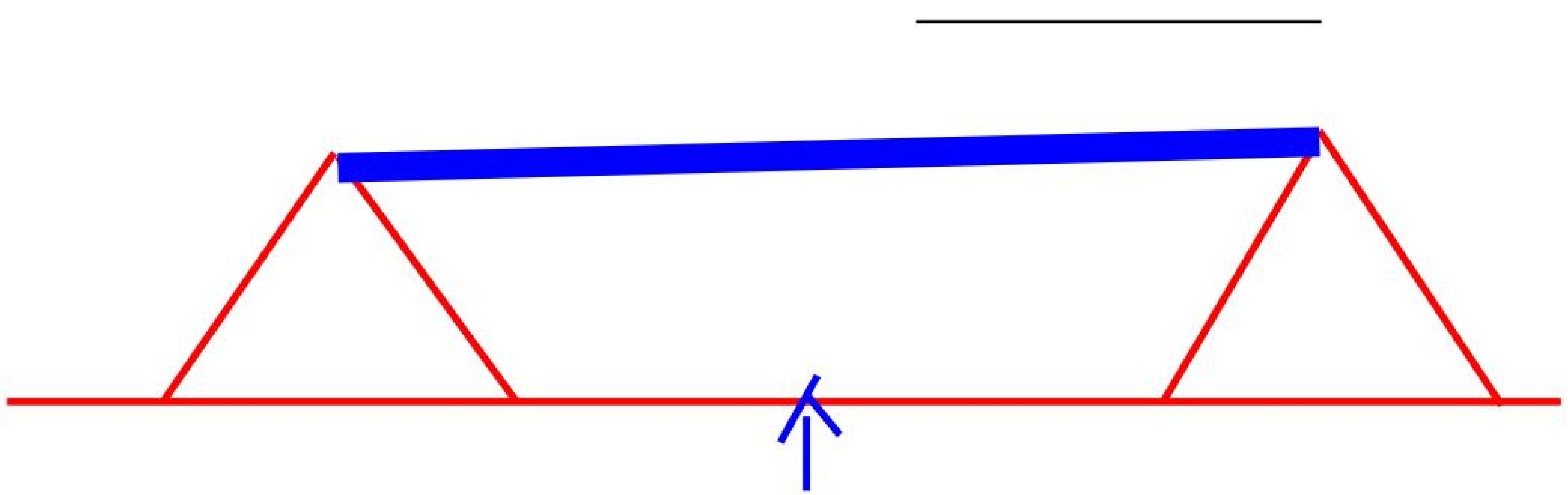


Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
- First/last of maxima
- Middle of maxima
  - Continuity: bad!
  - Disambiguity: only due to an additional criterion
  - Computational complexity: low
  - Plausibility: may be a problem

Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
- First/last of maxima
- Middle of maxima
  - Continuity: bad!
  - Disambiguity: only due to an additional criterion
  - Computational complexity: low
  - Plausibility: may be a problem



Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
- First/last of maxima
- Middle of maxima
- Any of maxima (chosen at random)
  - Continuity: bad!
  - Disambiguity: sometimes violated!
  - Computational complexity: low
  - Plausibility: reasonable
  - Can be applied to any form of consequents (not necessarily convex or even non-numerical)

Methods of defuzzification

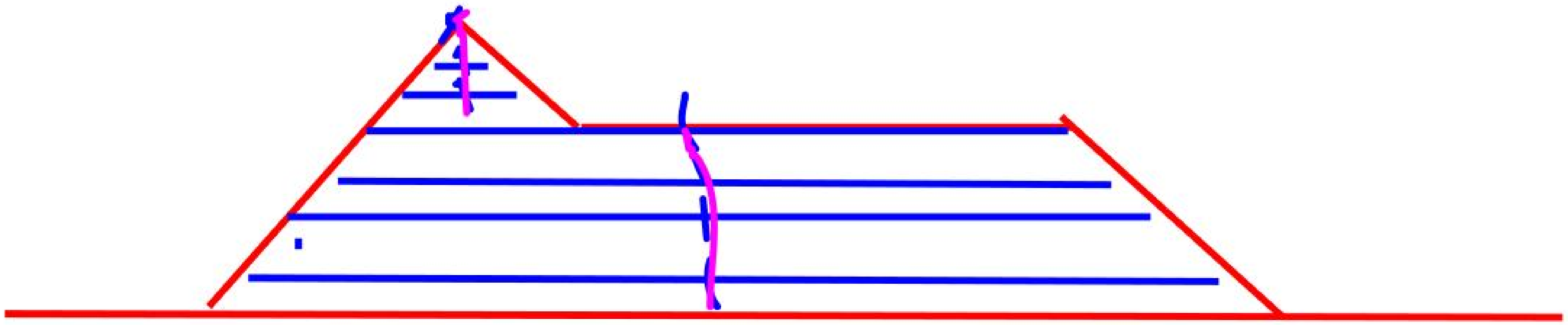
- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
- First/last of maxima
- Middle of maxima
- Any of maxima (chosen at random)
- Height defuzzification (each consequent is replaced by a singleton and their weighted mean is computed)
  - Continuity: good
  - Disambiguity: none
  - Computational complexity: low
  - Plausibility: doubtful!
  - Some features of fuzzy control are lost; in fact, crisp outputs of rules are combined

Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
- First/last of maxima
- Middle of maxima
- Any of maxima (chosen at random)
- Height defuzzification (each consequent is replaced by a singleton and their weighted mean is computed)

Methods of defuzzification

- Center of area (gravity) – ignores the multiplicity of overlapping consequents
- Center of sums – respects the multiplicity of overlapping consequents
- Center of largest area
- First/last of maxima
- Middle of maxima
- Any of maxima (chosen at random)
- Height defuzzification (each consequent is replaced by a singleton and their weighted mean is computed)



Defuzzification

Problems of defuzzification:

- Multiple maxima
- Continuous switching between rules
- If supports of consequents are not bounded, extending the universe may lead to different outputs

Takagi–Sugeno controller

Uses rules in a generalized form

if X is A_1 **then** Y is $f_1(X)$ **and**

$\dots$

if X is A_n **then** Y is $f_n(X)$

where f_i, $i = 1, \dots, n$, may be arbitrary functions of the input variables (usually linear)

Defuzzification

Problems of defuzzification:

- Multiple maxima
- Continuous switching between rules
- If supports of consequents are not bounded, extending the universe may lead to different outputs