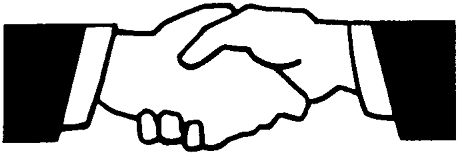
Einführung in den π-Kalkül

Christian Müller
Betreuer: Gert Smolka

5. Oktober 2004

Fahrplan

Was können wir bisher beschreiben?

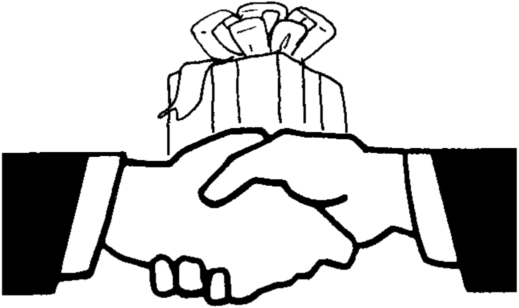


Handshake in CCS:

$$a.A \mid \overline{a}.B \;\rightarrow\; A \mid B$$

$\Rightarrow$ keine Möglichkeit Informationen zu übertragen

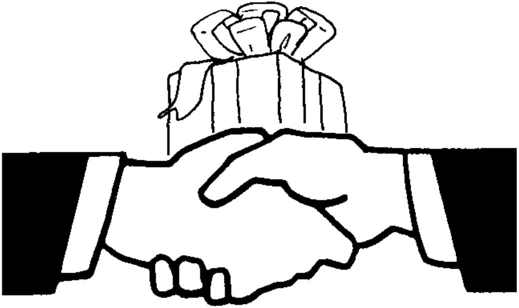
Übertragung im π-Kalkül



Was wird übertragen?

Wie findet Übertragung statt?

Übertragung im π-Kalkül



Was wird übertragen?

Namen

Wie findet Übertragung statt?

$$\bar{a}\langle x\rangle.A \mid a(y).B \;\longrightarrow\; A \mid \{{}^{x}/_{y}\}B$$

Syntax

Definition (π-Kalkül)

Die Menge $\mathcal{P}^\pi$ der π-Kalkül Prozessausdrücke ist definiert als:

$$P ::= \sum_{i \in I} \pi_i . P_i \quad | \quad P_1 | P_2 \quad | \quad \mathrm{new}\, x\, P \quad | \quad A\langle \vec{a} \rangle$$

Definition (Aktionspräfixe)

$$\begin{array}{lll} \pi & ::= & x(y) \quad \text{empfange } y \text{ über } x \\ & & \overline{x}\langle y \rangle \quad \text{sende } y \text{ über } x \\ & & \tau \quad \text{unsichtbare Aktion} \end{array}$$

Auf dem Weg zur strukturellen Kongruenz

Definition (Prozess-Kongruenz)

$\cong$ ist eine Prozess-Kongruenz, wenn:

- $\cong$ eine *Äquivalenzrelation* über $\mathcal{P}^\pi$ ist
- $\forall P, Q \in \mathcal{P}^\pi \, \forall \mathcal{C} : \; P \cong Q \;\Rightarrow\; \mathcal{C}[P] \cong \mathcal{C}[Q]$

Auf dem Weg zur strukturellen Kongruenz

Definition (Prozess-Kongruenz)

$\cong$ ist eine Prozess-Kongruenz, wenn:

- $\cong$ eine *Äquivalenzrelation* über $\mathcal{P}^\pi$ ist
- $\forall P, Q \in \mathcal{P}^\pi \; \forall \mathcal{C} : \; P \cong Q \; \Rightarrow \; \mathcal{C}[P] \cong \mathcal{C}[Q]$

Definition (Prozess-Kontext)

$$\mathcal{C} ::= [\,] \quad | \quad \pi.\mathcal{C} + M \quad | \quad \text{new}\, x\, \mathcal{C} \quad | \quad \mathcal{C}|P \quad | \quad P|\mathcal{C}$$

Strukurelle Kongruenz

Definition

$P \equiv Q$, wenn sie durch die folgenden Regeln syntaktisch gleich gemacht werden können:

Strukurelle Kongruenz

Definition

$P \equiv Q$, wenn sie durch die folgenden Regeln syntaktisch gleich gemacht werden können:

1. α-Umbennenung
2. Umordung in Summen und parallelen Kompositionen

Strukurelle Kongruenz

Definition

$P \equiv Q$, wenn sie durch die folgenden Regeln syntaktisch gleich gemacht werden können:

1. α-Umbennenung
2. Umordung in Summen und parallelen Kompositionen
3. $\text{new}\, x\, (P|Q) \equiv P|\text{new}\, x\, Q \qquad \text{falls } x \notin \text{fn}(P)$

Strukurelle Kongruenz

Definition

$P \equiv Q$, wenn sie durch die folgenden Regeln syntaktisch gleich gemacht werden können:

1. α-Umbennenung
2. Umordung in Summen und parallelen Kompositionen
3. $\mathsf{new}\, x\, (P|Q) \equiv P|\mathsf{new}\, x\, Q \qquad$ falls $x \notin \mathrm{fn}(P)$
4. $A(\vec{a}) \stackrel{def}{=} P_A \;\Rightarrow\; A\langle\vec{b}\rangle \equiv \{^{\vec{b}}/_{\vec{a}}\}P_A$

Strukurelle Kongruenz

Definition

$P \equiv Q$, wenn sie durch die folgenden Regeln syntaktisch gleich gemacht werden können:

1. α-Umbennenung
2. Umordung in Summen und parallelen Kompositionen
3. $\text{new}\, x\, (P|Q) \equiv P|\text{new}\, x\, Q \qquad \text{falls } x \notin \text{fn}(P)$
4. $A(\vec{a}) \overset{def}{=} P_A \;\Rightarrow\; A\langle\vec{b}\rangle \equiv \{^{\vec{b}}/_{\vec{a}}\}P_A$
5. $P|0 \equiv P \qquad \text{new}\, x\, 0 \equiv 0 \qquad \text{new}\, xy\, P \equiv \text{new}\, yx\, P$

Reaktion

$$\frac{Q \equiv P \qquad P \rightarrow P' \qquad P' \equiv Q'}{Q \rightarrow Q'} \qquad (\textsc{Struct})$$

Reaktion

$$\frac{Q \equiv P \qquad P \rightarrow P' \qquad P' \equiv Q'}{Q \rightarrow Q'} \qquad (\textsc{Struct})$$

$$(x(y).P + M)|(\overline{x}\langle z\rangle.Q + N) \rightarrow \{^z/_y\}P|Q \qquad (\textsc{React})$$

Reaktion

$$\frac{Q \equiv P \qquad P \rightarrow P' \qquad P' \equiv Q'}{Q \rightarrow Q'} \qquad (\textsc{Struct})$$

$$(x(y).P + M)|(\overline{x}\langle z\rangle.Q + N) \rightarrow \{^z/_y\}P|Q \qquad (\textsc{React})$$

$$\tau.P + M \rightarrow P \qquad (\textsc{Tau})$$

Reaktion

$$\frac{Q \equiv P \qquad P \rightarrow P' \qquad P' \equiv Q'}{Q \rightarrow Q'} \qquad (\textsc{Struct})$$

$$(x(y).P + M) | (\overline{x}\langle z\rangle.Q + N) \rightarrow \{^z/_y\}P|Q \qquad (\textsc{React})$$

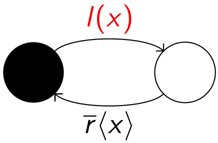
$$\tau.P + M \rightarrow P \qquad (\textsc{Tau})$$

$$\frac{P \rightarrow P'}{P|Q \rightarrow P'|Q} \qquad (\textsc{Par})$$

$$\frac{P \rightarrow P'}{\mathrm{new}\, x\, P \rightarrow \mathrm{new}\, x\, P'} \qquad (\textsc{Res})$$

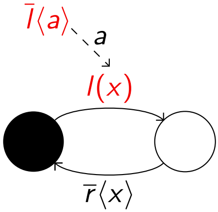
Pufferzelle

$$B(l, r) \stackrel{def}{=} l(x).\overline{r}\langle x\rangle.B\langle l, r\rangle$$



Pufferzelle

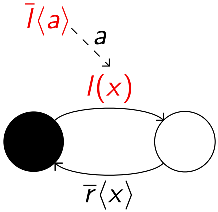
$$B(l, r) \stackrel{def}{=} l(x).\overline{r}\langle x\rangle.B\langle l, r\rangle$$

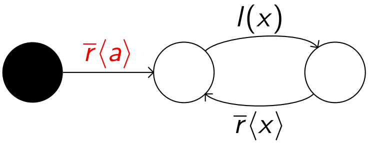


$$\bar{l}\langle a\rangle \mid l(x).\overline{r}\langle x\rangle.B \;\rightarrow\; 0 \mid \{^{a}/_{x}\}(\overline{r}\langle x\rangle.B)$$

Pufferzelle

$$B(l, r) \stackrel{def}{=} l(x).\bar{r}\langle x\rangle.B\langle l, r\rangle \qquad\qquad \bar{r}\langle a\rangle.B\langle l, r\rangle$$





$$\bar{l}\langle a\rangle \mid l(x).\bar{r}\langle x\rangle.B \;\rightarrow\; 0 \mid \{^{a}/_{x}\}(\bar{r}\langle x\rangle.B)$$

Pufferzelle in Aktion - formale Betrachtung

$$
\begin{aligned}
B(l, r) &\stackrel{def}{=} l(x).\overline{r}\langle x\rangle.B\langle l, r\rangle \\
P &\stackrel{def}{=} \overline{l_1}\langle m\rangle \mid \\
&\quad B\langle l_1, r_1\rangle \mid B\langle l_2, r_2\rangle \mid \\
&\quad r_1(y).P_1 + r_2(z).P_2
\end{aligned}
$$

Pufferzelle in Aktion - formale Betrachtung

$$
\begin{aligned}
B(l,r) &\stackrel{def}{=} l(x).\overline{r}\langle x\rangle.B\langle l,r\rangle \\
P &\stackrel{def}{=} \overline{l_1}\langle m\rangle \mid \\
&\phantom{\stackrel{def}{=}} B\langle l_1,r_1\rangle \mid B\langle l_2,r_2\rangle \mid \\
&\phantom{\stackrel{def}{=}} r_1(y).P_1 + r_2(z).P_2
\end{aligned}
$$

$$
P \quad \rightarrow^{\star} \quad \overline{l_1}\langle m\rangle \mid l_1(x).\overline{r_1}\langle x\rangle.B\langle l_1,r_1\rangle \mid B\langle l_2,r_2\rangle \mid (r_1(y).P_1 + r_2(z).P_2)
$$

Pufferzelle in Aktion - formale Betrachtung

$$
\begin{aligned}
B(l,r) &\stackrel{def}{=} l(x).\overline{r}\langle x\rangle.B\langle l,r\rangle \\
P &\stackrel{def}{=} \overline{l_1}\langle m\rangle \mid \\
&\quad\; B\langle l_1,r_1\rangle \mid B\langle l_2,r_2\rangle \mid \\
&\quad\; r_1(y).P_1 + r_2(z).P_2
\end{aligned}
$$

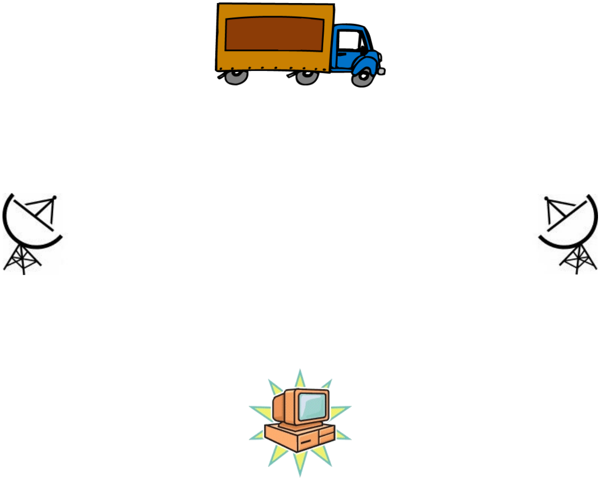
$$
\begin{aligned}
P &\rightarrow^\star \overline{l_1}\langle m\rangle \mid l_1(x).\overline{r_1}\langle x\rangle.B\langle l_1,r_1\rangle \mid B\langle l_2,r_2\rangle \mid (r_1(y).P_1 + r_2(z).P_2) \\
&\rightarrow^\star \overline{r_1}\langle m\rangle.B\langle l_1,r_1\rangle \mid B\langle l_2,r_2\rangle \mid (r_1(y).P_1 + r_2(z).P_2)
\end{aligned}
$$

Pufferzelle in Aktion - formale Betrachtung

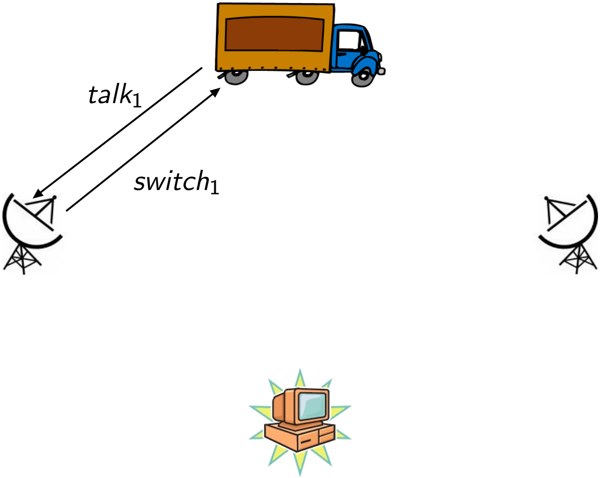
$$
\begin{aligned}
B(l,r) &\stackrel{def}{=} l(x).\overline{r}\langle x\rangle.B\langle l,r\rangle \\
P &\stackrel{def}{=} \overline{l_1}\langle m\rangle \mid \\
&\quad\; B\langle l_1,r_1\rangle \mid B\langle l_2,r_2\rangle \mid \\
&\quad\; r_1(y).P_1 + r_2(z).P_2
\end{aligned}
$$

$$
\begin{aligned}
P &\rightarrow^\star \overline{l_1}\langle m\rangle \mid l_1(x).\overline{r_1}\langle x\rangle.B\langle l_1,r_1\rangle \mid B\langle l_2,r_2\rangle \mid (r_1(y).P_1 + r_2(z).P_2) \\
&\rightarrow^\star \overline{r_1}\langle m\rangle.B\langle l_1,r_1\rangle \mid B\langle l_2,r_2\rangle \mid (r_1(y).P_1 + r_2(z).P_2) \\
&\rightarrow^\star B\langle l_1,r_1\rangle \mid B\langle l_2,r_2\rangle \mid \{^m/_y\}P_1
\end{aligned}
$$

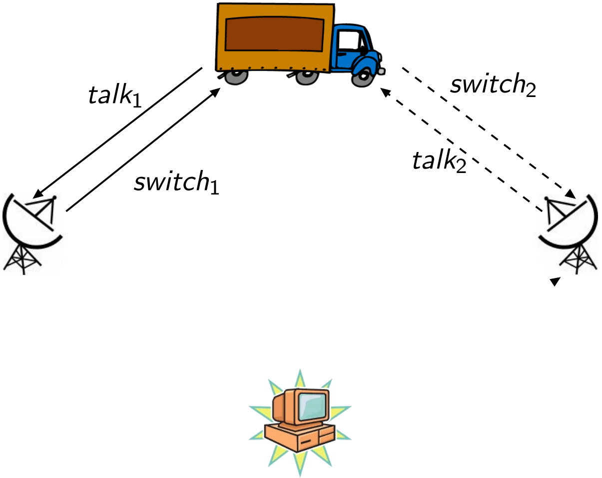
Mautsystem



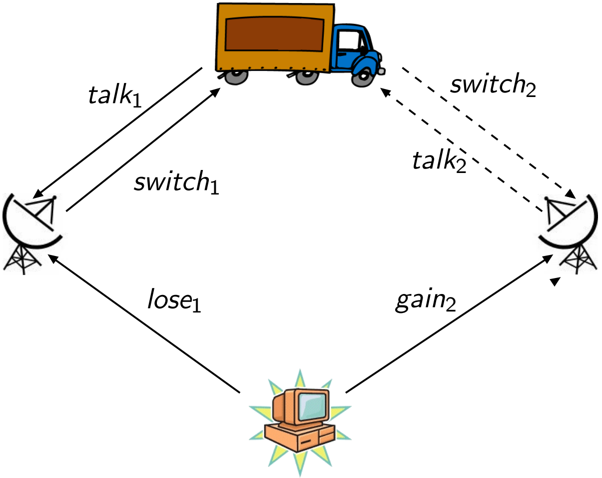
Mautsystem



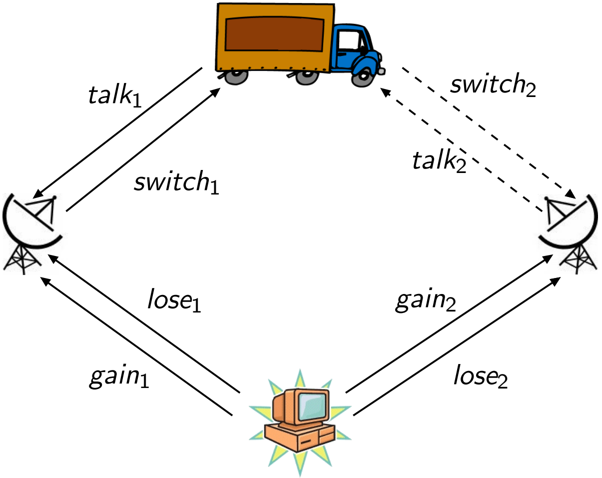
Mautsystem



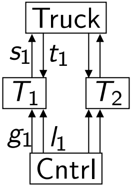
Mautsystem



Mautsystem



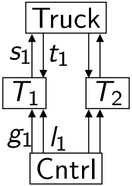
Mautsystem in CCS



$$Truck_1 \stackrel{def}{=} \overline{talk_1}.Truck_1 + switch_1.Truck_2$$

$$Truck_2 \stackrel{def}{=} \overline{talk_2}.Truck_2 + switch_2.Truck_1$$

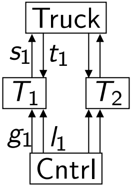
Mautsystem in CCS



$$Truck_1 \stackrel{def}{=} \overline{talk_1}.Truck_1 + switch_1.Truck_2$$
$$Truck_2 \stackrel{def}{=} \overline{talk_2}.Truck_2 + switch_2.Truck_1$$

$$Trans_1 \stackrel{def}{=} talk_1.Trans_1 + lose_1.\overline{switch_1}.gain_1.Trans_1$$

Mautsystem in CCS

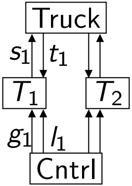


$$
\begin{aligned}
Truck_1 &\stackrel{def}{=} \overline{talk_1}.Truck_1 + switch_1.Truck_2 \\
Truck_2 &\stackrel{def}{=} \overline{talk_2}.Truck_2 + switch_2.Truck_1
\end{aligned}
$$

$$
Trans_1 \stackrel{def}{=} talk_1.Trans_1 + lose_1.\overline{switch_1}.gain_1.Trans_1
$$

$$
Cntrl \stackrel{def}{=} \overline{lose_1}.\overline{gain_2}.\overline{lose_2}.\overline{gain_1}.Cntrl
$$

Mautsystem in CCS



$$Truck_1 \stackrel{def}{=} \overline{talk_1}.Truck_1 + switch_1.Truck_2$$

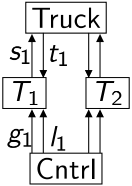
$$Truck_2 \stackrel{def}{=} \overline{talk_2}.Truck_2 + switch_2.Truck_1$$

$$Trans_1 \stackrel{def}{=} talk_1.Trans_1 + lose_1.\overline{switch_1}.gain_1.Trans_1$$

$$Cntrl \stackrel{def}{=} \overline{lose_1}.\overline{gain_2}.\overline{lose_2}.\overline{gain_1}.Cntrl$$

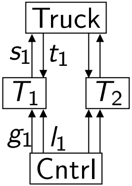
$$System_1 \stackrel{def}{=} Truck_1 | Trans_1 | Cntrl | gain_2.Trans_2$$

Mautsystem im π-Kalkül



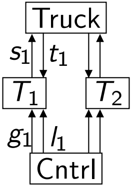
$$Truck_1 \stackrel{def}{=} \overline{talk_1}.Truck_1 + switch_1.Truck_2$$
$$Truck_2 \stackrel{def}{=} \overline{talk_2}.Truck_2 + switch_2.Truck_1$$

Mautsystem im π-Kalkül



$$
\begin{aligned}
Truck(talk, switch) \stackrel{def}{=} \; & \overline{talk}.Truck\langle talk, switch\rangle + \\
& switch(t, s).Truck\langle t, s\rangle
\end{aligned}
$$

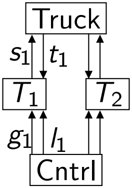
Mautsystem im π-Kalkül



$$\begin{aligned} Truck(talk, switch) &\stackrel{def}{=} \overline{talk}.Truck\langle talk, switch\rangle + \\ & \quad switch(t, s).Truck\langle t, s\rangle \end{aligned}$$

$$Trans_1 \stackrel{def}{=} talk_1.Trans_1 + lose_1.\overline{switch_1}.gain_1.Trans_1$$

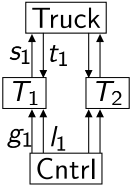
Mautsystem im π-Kalkül



$$
\begin{aligned}
Truck(talk, switch) &\overset{def}{=} \overline{talk}.Truck\langle talk, switch\rangle + \\
&\quad switch(t, s).Truck\langle t, s\rangle
\end{aligned}
$$

$$
\begin{aligned}
Trans_1(talk, switch) &\overset{def}{=} talk.Trans_1\langle talk, switch\rangle + \\
&\quad lose_1(t, s).\overline{switch}\langle t, s\rangle.gain_1(t, s).Trans_1\langle t, s\rangle
\end{aligned}
$$

Mautsystem im π-Kalkül

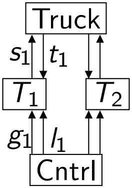


$$Truck(talk, switch) \stackrel{def}{=} \overline{talk}.Truck\langle talk, switch\rangle + switch(t,s).Truck\langle t,s\rangle$$

$$Trans_1(talk, switch) \stackrel{def}{=} talk.Trans_1\langle talk, switch\rangle + lose_1(t,s).\overline{switch}\langle t,s\rangle.gain_1(t,s).Trans_1\langle t,s\rangle$$

$$Cntrl \stackrel{def}{=} \overline{lose_1}.\overline{gain_2}.\overline{lose_2}.\overline{gain_1}.Cntrl$$

Mautsystem im π-Kalkül

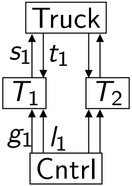


$$
\begin{aligned}
Truck(talk, switch) &\stackrel{def}{=} \overline{talk}.Truck\langle talk, switch\rangle + \\
&\quad switch(t, s).Truck\langle t, s\rangle
\end{aligned}
$$

$$
\begin{aligned}
Trans_1(talk, switch) &\stackrel{def}{=} talk.Trans_1\langle talk, switch\rangle + \\
&\quad lose_1(t, s).\overline{switch}\langle t, s\rangle.gain_1(t, s).Trans_1\langle t, s\rangle
\end{aligned}
$$

$$
\begin{aligned}
Cntrl(l) &\stackrel{def}{=} \text{new } t\ s\ \bar{l}\langle t, s\rangle. \\
&\quad (\overline{gain_1}\langle t, s\rangle.Cntrl\langle lose_1\rangle + \overline{gain_2}\langle t, s\rangle.Cntrl\langle lose_2\rangle)
\end{aligned}
$$

Mautsystem im π-Kalkül



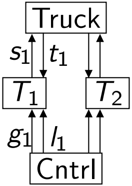
$$Truck(talk, switch) \stackrel{def}{=} \overline{talk}.Truck\langle talk, switch\rangle + switch(t, s).Truck\langle t, s\rangle$$

$$Trans_1(talk, switch) \stackrel{def}{=} talk.Trans_1\langle talk, switch\rangle + lose_1(t, s).\overline{switch}\langle t, s\rangle.gain_1(t, s).Trans_1\langle t, s\rangle$$

$$Cntrl(l) \stackrel{def}{=} \text{new } t\ s\ \bar{l}\langle t, s\rangle.(\overline{gain_1}\langle t, s\rangle.Cntrl\langle lose_1\rangle + \overline{gain_2}\langle t, s\rangle.Cntrl\langle lose_2\rangle)$$

$$System_1 \stackrel{def}{=} Truck_1|Trans_1|Cntrl|gain_2.Trans_2$$

Mautsystem im π-Kalkül



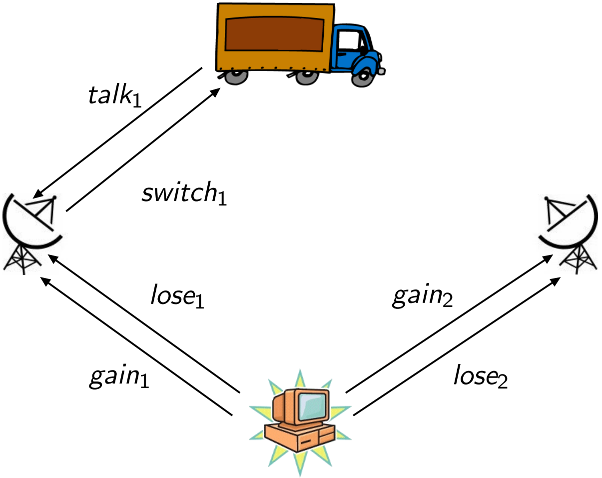
$$
\begin{aligned}
Truck(talk, switch) &\stackrel{def}{=} \overline{talk}.Truck\langle talk, switch\rangle + \\
&\quad switch(t, s).Truck\langle t, s\rangle
\end{aligned}
$$

$$
\begin{aligned}
Trans_1(talk, switch) &\stackrel{def}{=} talk.Trans_1\langle talk, switch\rangle + \\
&\quad lose_1(t, s).\overline{switch}\langle t, s\rangle.gain_1(t, s).Trans_1\langle t, s\rangle
\end{aligned}
$$

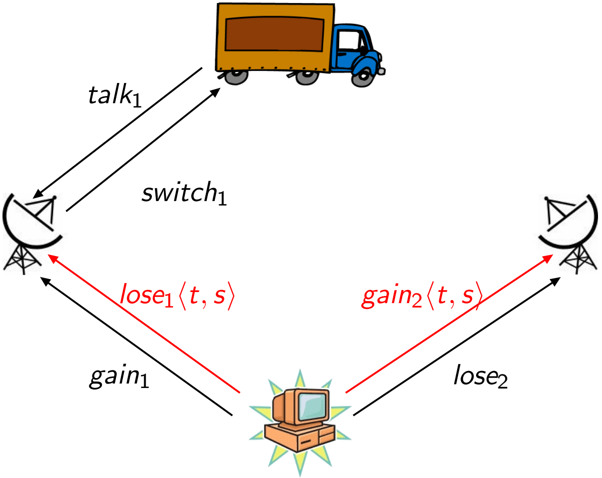
$$
\begin{aligned}
Cntrl(l) &\stackrel{def}{=} \text{new } t\ s\ \bar{l}\langle t, s\rangle. \\
&\quad (\overline{gain_1}\langle t, s\rangle.Cntrl\langle lose_1\rangle + \overline{gain_2}\langle t, s\rangle.Cntrl\langle lose_2\rangle)
\end{aligned}
$$

$$
\begin{aligned}
System_1 &\stackrel{def}{=} Truck\langle t_1, s_1\rangle | Trans_1\langle t_1, s_1\rangle | Cntrl\langle lose_1\rangle \\
&\quad | gain_2(t, s).Trans_2\langle t, s\rangle
\end{aligned}
$$

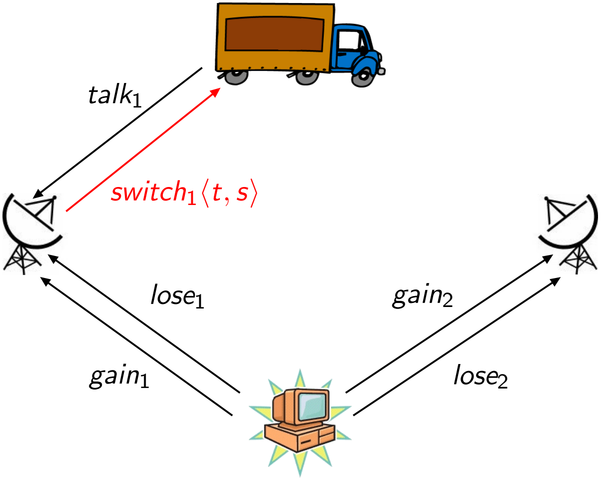
Mautsystem im π-Kalkül



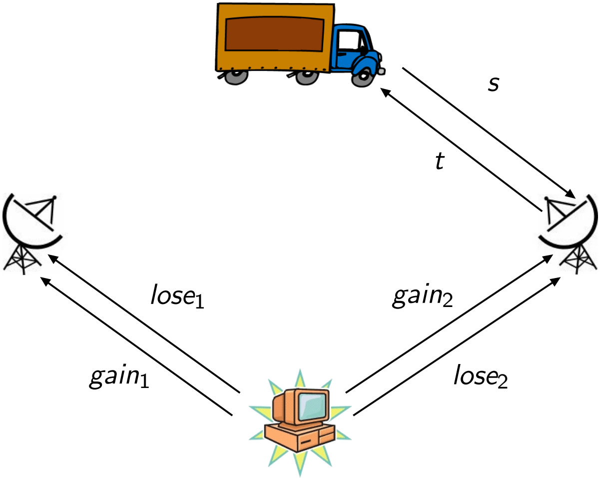
Mautsystem im π-Kalkül



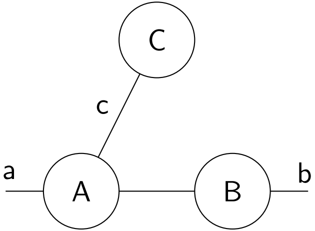
Mautsystem im π-Kalkül



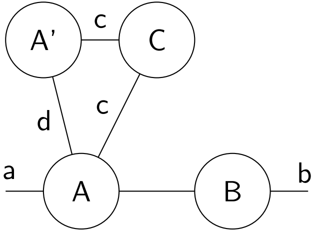
Mautsystem im π-Kalkül



Mobilität

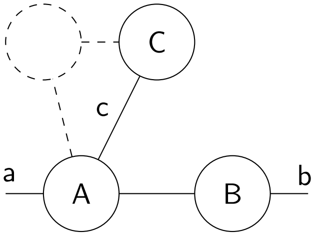


Mobilität



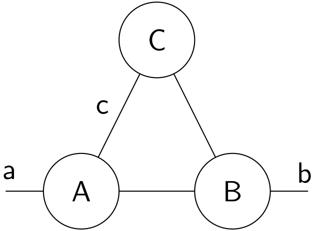
- neue Komponenten erzeugen

Mobilität



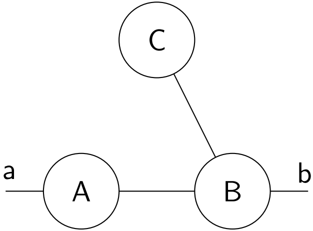
- neue Komponenten erzeugen
- bestehendende Komponenten entfernen

Mobilität



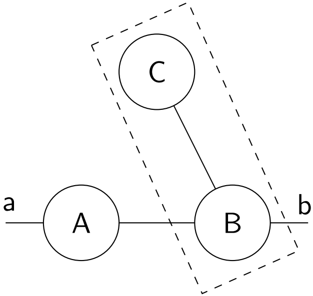
- neue Komponenten erzeugen
- bestehendende Komponenten entfernen
- π-Kalkül:
  neue Verbindung zwischen bestehenden Komponenten schaffen

Mobilität



- neue Komponenten erzeugen
- bestehendende Komponenten entfernen
- π-Kalkül:
  neue Verbindung zwischen bestehenden Komponenten schaffen

Mobilität



- neue Komponenten erzeugen
- bestehendende Komponenten entfernen
- π-Kalkül:
  neue Verbindung zwischen bestehenden Komponenten schaffen

Vereinfachungen möglich?

- rekursive Definitionen nützlich zum Bauen von Systemen, aber unhandlich für Beweise
- Austausch von Namen ermöglicht kompakteren Formalismus

Vereinfachungen möglich?

- rekursive Definitionen nützlich zum Bauen von Systemen, aber unhandlich für Beweise
- Austausch von Namen ermöglicht kompakteren Formalismus

Replikation:

Prozessausdrücke $P ::= \ldots \mid\, !P$

Strukturelle Kongruenz: $!P \equiv P | !P$

Was fehlt noch?

- polyadischer π-Kalkül
- Kodierung rekursiver Definitionen mit Replikation
- Beschreibung von Verhalten
- Interaktion mit der Außenwelt

Referenzen

- Milner R., Parrow, J. and Walker D., *A calculus of mobile processes, Parts I and II*, Information and Computation, 100, 1, 1992, pp1-77
- Milner, R., *communication and mobile systems: the π-calculus*, Cambridge University Press, 1999
- Sangiorgi, D. and Walker, D., *The π-calculus: A Theory of Mobile Processes*, Cambridge University Press, 2001
- Milner, R., *What's in a name*, January 2003, `http://www.cl.cam.ac.uk/users/rm135/wosname.pdf`