Algumas notas da disciplina de Teoria dos Processos Concorrentes LMAC

Pedro Resende

Departamento de Matemática
Instituto Superior Técnico

Conteúdo

Capítulo 1

Notas de 1997/98

1.1 Sistemas de transição, equivalências e processos

No que se segue Act será sempre um conjunto, cujos elementos designaremos por *acções*.

1.1.1

Um *sistema de transição (st) sobre* Act, $\mathsf{S} = \langle P, T \rangle$, consiste num conjunto P, de *estados*, e num conjunto $T \subseteq P \times Act \times P$, de *transições*. O conjunto T é também denominado *relação de transição*. Utilizaremos a seguinte notação:

$$\begin{aligned} x \xrightarrow{\alpha} y &\iff^{\text{def}} \langle x, \alpha, y \rangle \in T \ , \\ x \xrightarrow{\alpha} &\iff^{\text{def}} \exists_{y \in P} (x \xrightarrow{\alpha} y) \ . \end{aligned}$$

Um abuso frequente de linguagem consistirá em identificar um st com o seu conjunto de estados sempre que não houver possibilidade de confusão no que respeita à relação de transição. Por exemplo, poderíamos referir-nos a S como o "sistema P".

1.1.2

Um *traço* (sobre Act) é uma sequência finita de acções, sendo o conjunto dos traços sobre Act representado por Act^*. Dado um traço $t = \alpha_1 \ldots \alpha_n$, n é dito o *comprimento* do traço. O traço de comprimento zero é representado por ε.

Seja $\langle P, T\rangle$ um st sobre Act. A relação de transição pode ser estendida a traços como sendo o conjunto $T' \subseteq P \times Act^* \times P$ tal que

$$\begin{aligned} \langle x, \varepsilon, y\rangle \in T' &\iff x = y \ , \\ \langle x, \alpha t, y\rangle \in T' &\iff \exists_{z\in P}((x \xrightarrow{\alpha} z) \wedge (\langle z, t, y\rangle \in T')) \ . \end{aligned}$$

Utilizaremos também a notação $x \xrightarrow{t} y$, $x \xrightarrow{t}$, etc., para indicar $\langle x, t, y\rangle \in T'$, etc.

1.1.3

Um st *apontado (sta)*, $\langle P, T, \imath\rangle$, é um st $\langle P, T\rangle$ no qual se distingue um estado $\imath \in P$, denominado *estado inicial*.

Um sta diz-se *acessível* se para qualquer estado $x \in P$ existe um traço $t \in Act^*$ tal que $\imath \xrightarrow{t} x$ (i.e., x é acessível a partir do estado inicial).

Sistemas de transição apontados e acessíveis (abrev. staas) serão usados para representar o comportamento de sistemas concorrentes. A cada staa corresponde uma máquina capaz de executar acções de Act, cujos estados são os elementos de P e cujo estado inicial é $\imath$.

1.1.4

Sejam $\langle P, T\rangle$ e $\langle Q, U\rangle$ sts. Um *morfismo* $f : \langle P, T\rangle \to \langle Q, U\rangle$ é uma função $f : P \to Q$ tal que para todo $x, y \in P$, $\alpha \in Act$,

$$x \xrightarrow{\alpha} y \Rightarrow f(x) \xrightarrow{\alpha} f(y) \ .$$

Um morfismo f é um *isomorfismo* se for uma bijecção e f^{-1} for também um morfismo.

Proposição. *$f : \langle P, T\rangle \to \langle Q, U\rangle$ é um isomorfismo sse for bijectiva e para todo $x, y \in P$, $\alpha \in Act$,*

$$x \xrightarrow{\alpha} y \iff f(x) \xrightarrow{\alpha} f(y) \ .$$

Dizemos que dois sistemas S e T são *isomorfos*, e escrevemos $\mathsf{S} \cong \mathsf{T}$, quando existe um isomorfismo $f : \mathsf{S} \to \mathsf{T}$.

Exercício. Mostre que a relação de isomorfismo é de equivalência.

Um *morfismo de stas*, $f : \langle P, T, \imath\rangle \to \langle Q, U, \jmath\rangle$, é um morfismo de sts que preserva o estado inicial; isto é, tal que $f(\imath) = \jmath$. Os resultados anteriores generalizam-se de forma óbvia a este caso.

1.1.5

Seja $\mathsf{S} = \langle P, T, \imath \rangle$ um staa. Usaremos a notação $\mathcal{T}(\mathsf{S})$ para o conjunto de traços "executáveis"por S, i.e.,

$$\mathcal{T}(\mathsf{S}) \stackrel{\text{def}}{=} \{t \in Act^* \mid \imath \xrightarrow{t}\} \ .$$

Dois staas S e T dizem-se *equivalentes por traços*, e escrevemos $\mathsf{S} \sim_{\mathcal{T}} \mathsf{T}$, se $\mathcal{T}(\mathsf{S}) = \mathcal{T}(\mathsf{T})$.

1.1.6

Seja $\mathsf{S} = \langle P, T, \imath \rangle$ um staa, e $x \in P$. Defina-se a seguinte notação:

$$\begin{array}{rcl}
\mathsf{r}(x) & \stackrel{\text{def}}{=} & \{\alpha \in Act \mid x \xrightarrow{\alpha}\} \ , \\
\mathsf{f}(x) & \stackrel{\text{def}}{=} & Act \setminus \mathsf{r}(x) \ , \\
\mathcal{CT}(\mathsf{S}) & \stackrel{\text{def}}{=} & \{t \in Act^* \mid \exists_{y \in P}((\imath \xrightarrow{t} y) \wedge (\mathsf{r}(y) = \emptyset))\} \ , \\
\mathcal{F}(\mathsf{S}) & \stackrel{\text{def}}{=} & \{\langle t, X\rangle \in Act^* \times 2^{Act} \mid \exists_{y \in P}((\imath \xrightarrow{t} y) \wedge (X \subseteq \mathsf{f}(y)))\} \ , \\
\mathcal{F}_{\text{fin}}(\mathsf{S}) & \stackrel{\text{def}}{=} & \{\langle t, X\rangle \in Act^* \times 2^{Act}_{\text{fin}} \mid \exists_{y \in P}((\imath \xrightarrow{t} y) \wedge (X \subseteq \mathsf{f}(y)))\} \ , \\
\mathcal{R}(\mathsf{S}) & \stackrel{\text{def}}{=} & \{\langle t, X\rangle \in Act^* \times 2^{Act} \mid \exists_{y \in P}((\imath \xrightarrow{t} y) \wedge (X = \mathsf{r}(y)))\} \ , \\
\mathcal{FT}(\mathsf{S}) & \stackrel{\text{def}}{=} & \{\langle X_0, \alpha_1, \ldots, X_n\rangle \mid \\
 & & \exists_{x_0,\ldots,x_n}((\imath = x_0 \xrightarrow{\alpha_1} x_1 \xrightarrow{\alpha_2} \cdots \xrightarrow{\alpha_n} x_n) \wedge (X_i \subseteq \mathsf{f}(x_i)\ (0 \leq i \leq n)))\} \ , \\
\mathcal{FT}_{\text{fin}}(\mathsf{S}) & \stackrel{\text{def}}{=} & \{\langle X_0, \alpha_1, \ldots, X_n\rangle \mid \\
 & & \exists_{x_0,\ldots,x_n}((\imath = x_0 \xrightarrow{\alpha_1} x_1 \xrightarrow{\alpha_2} \cdots \xrightarrow{\alpha_n} x_n) \wedge (X_i \subseteq_{\text{fin}} \mathsf{f}(x_i)\ (0 \leq i \leq n)))\} \ , \\
\mathcal{RT}(\mathsf{S}) & \stackrel{\text{def}}{=} & \{\langle X_0, \alpha_1, \ldots, X_n\rangle \mid \\
 & & \exists_{x_0,\ldots,x_n}((\imath = x_0 \xrightarrow{\alpha_1} x_1 \xrightarrow{\alpha_2} \cdots \xrightarrow{\alpha_n} x_n) \wedge (X_i = \mathsf{r}(x_i)\ (0 \leq i \leq n)))\} \ .
\end{array}$$

Para cada $E \in \{\mathcal{T}, \mathcal{CT}, \mathcal{F}, \mathcal{F}_{\text{fin}}, \mathcal{FT}, \mathcal{FT}_{\text{fin}}, \mathcal{R}, \mathcal{RT}\}$ defina-se agora a equivalência $\sim_E$ tal que

$$\mathsf{S} \sim_E \mathsf{T} \stackrel{\text{def}}{\iff} E(\mathsf{S}) = E(\mathsf{T}) \ .$$

Teorema. *As equivalências $\sim_E$ relacionam-se do seguinte modo:*

1. $\sim_{\mathcal{F}} \subsetneq \sim_{\mathcal{T}}$,
2. $\sim_{\mathcal{F}} \subsetneq \sim_{\mathcal{CT}}$,

3. $\sim_{\mathcal{CT}}$ *e* $\sim_{\mathcal{T}}$ *são incomparáveis,*

4. $\sim_{\mathcal{FT}} \subsetneq \sim_{\mathcal{F}}$,

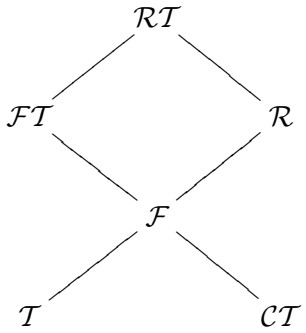
5. $\sim_{\mathcal{R}} \subsetneq \sim_{\mathcal{F}}$,

6. $\sim_{\mathcal{RT}} \subsetneq \sim_{\mathcal{R}}$,

7. $\sim_{\mathcal{RT}} \subsetneq \sim_{\mathcal{FT}}$,

8. $\sim_{\mathcal{FT}}$ *e* $\sim_{\mathcal{R}}$ *são incomparáveis.*

A situação descrita no teorema pode representar-se por meio do seguinte diagrama:



Exercício. Mostre que a intersecção $\sim_{\mathcal{T}} \cap \sim_{\mathcal{CT}}$ coincide com a equivalência $\sim_{\mathcal{CT}'}$ definida por

$$\mathsf{S} \sim_{\mathcal{CT}'} \mathsf{T} \overset{\text{def}}{\iff} \mathcal{CT}'(\mathsf{S}) = \mathcal{CT}'(\mathsf{T}) \ ,$$

onde $\mathcal{CT}'(\mathsf{S}) \subseteq Act^* \times \{0,1\}$ e

$$\begin{aligned} \langle t, 0\rangle \in \mathcal{CT}'(\mathsf{S}) &\iff t \in \mathcal{T}(\mathsf{S}) \ , \\ \langle t, 1\rangle \in \mathcal{CT}'(\mathsf{S}) &\iff t \in \mathcal{CT}(\mathsf{S}) \end{aligned}$$

[i.e., $\mathcal{CT}'(\mathsf{S}) = (\mathcal{T}(\mathsf{S}) \times \{0\}) \cup (\mathcal{CT}(\mathsf{S}) \times \{1\})$].

Nota. É habitual na literatura definir *equivalência por traços completos* como sendo $\sim_{\mathcal{CT}'}$ em vez de $\sim_{\mathcal{CT}}$.

1.1.7

Informalmente, chamamos *processo* ao comportamento *observável* (de acordo com alguma noção de observação) dum staa. Seja $\sim_E$ uma equivalência de staas (E pode ser, e.g., $\mathcal{T}$, $\mathcal{CT}$, etc.). A ideia por detrás duma tal equivalência é a de que dois staas são equivalentes quando exibem o mesmo comportamento observável. Portanto podemos definir *processo* genericamente como sendo uma classe de equivalência de staas (dizemos que um processo é um staa "módulo E"). Existem assim diversas definições de processo, dependendo da equivalência considerada.

Esta noção genérica pode não ser a mais útil na prática. Por exemplo, em geral as classes de equivalência de staas não são conjuntos (porquê?). Porém, em cada caso particular é usualmente possível obter uma representação mais conveniente. Cada representação será designada por *modelo de processos*.

Exemplo. No caso da equivalência de traços podemos identificar os processos com os conjuntos não vazios de traços, fechados para prefixos, i.e., aqueles subconjuntos não vazios X de Act^* para os quais se $t \in X$ e $t = su$ então $s \in X$ (v. exercício 1.1.8-8). Note-se que por esta definição tem-se sempre $\varepsilon \in X$; em particular, $\{\varepsilon\}$ é um processo (módulo $\sim_{\mathcal{T}}$). Esta representação diz-se *concreta* porque os processos são representados por conjuntos. Designaremos o modelo de processos assim obtido por *modelo de traços* e representá-lo-emos por **T**.

1.1.8 Exercícios

1. Seja $\mathsf{S} = \langle P, T\rangle$ um st, $x, y \in P$ e $s, t \in Act^*$. Prove que

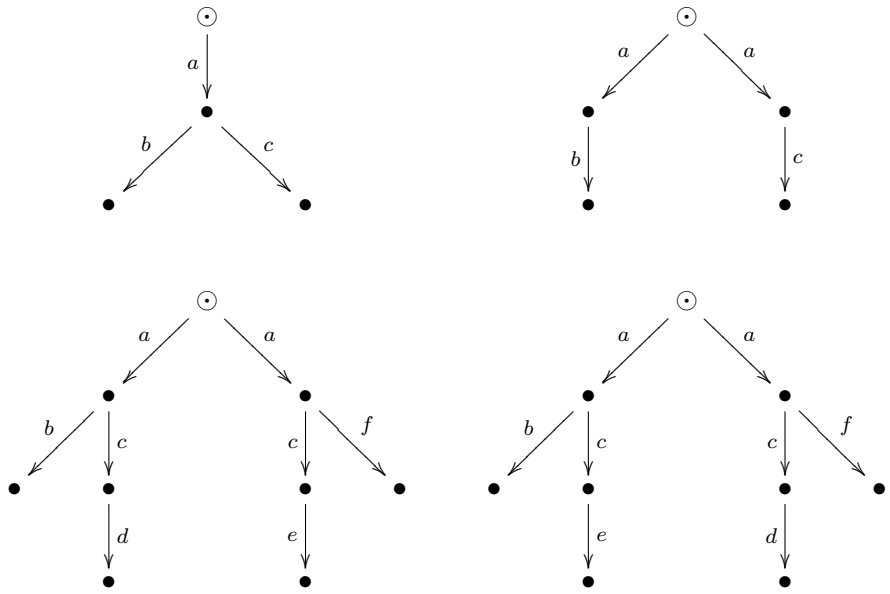
$$x \xrightarrow{st} y \iff \exists_{z \in P}(x \xrightarrow{s} z \text{ e } z \xrightarrow{t} y) .$$

2. Prove que se $f : \mathsf{S} \to \mathsf{T}$ é um morfismo de sts então tem-se, para qualquer $t \in Act^*$ e $x, y \in P$,

$$x \xrightarrow{t} y \Rightarrow f(x) \xrightarrow{t} f(y) .$$

3. Para cada um dos seguintes staas, calcule os conjuntos de traços ($\mathcal{T}$), traços completos ($\mathcal{CT}$), falhas ($\mathcal{F}$), traços de falha ($\mathcal{FT}$), menus ($\mathcal{R}$) e traços de menus ($\mathcal{RT}$). Para cada par destes sistemas diga também

quais das equivalências que estudou os identificam.



Nota: para $\mathcal{F}$ e $\mathcal{FT}$ assuma $Act = \{a, b, c, d, e, f\}$ (e $a \neq b$, $a \neq c$, $b \neq c$, etc.).

4. Prove o Teorema 1.1.6.

5. Mostre que $\sim_{\mathcal{CT}}$ e $\sim_{\mathcal{FT}_{\text{fin}}}$, assim como $\sim_{\mathcal{F}}$ e $\sim_{\mathcal{FT}_{\text{fin}}}$, são incomparáveis.

6. Mostre que $\cong \subsetneq \sim_{\mathcal{RT}}$.

7. Complete o diagrama de equivalências por forma a incluir $\mathcal{CT}'$, $\mathcal{F}_{\text{fin}}$, $\mathcal{FT}_{\text{fin}}$ e $\cong$.

8. (a) Seja S um staa arbitrário. Mostre que $\mathcal{T}(\mathsf{S})$ é não vazio e fechado para prefixos.

   (b) Seja $X \subseteq Act^*$ um conjunto não vazio e fechado para prefixos. Mostre que existe um staa S tal que $\mathcal{T}(\mathsf{S}) = X$. [Sugestão: tome X para conjunto de estados e defina $t \xrightarrow{\alpha} u$ se $u = t\alpha$, com estado inicial ε.]

9. Obtenha um modelo concreto de processos para $\mathcal{CT}'$.

1.2 Álgebras de processos

1.2.1

Começamos por reconhecer a existência de alguma estrutura algébrica na classe dos staas. Doravante designaremos esta classe por STAA.

Nota. STAA é uma classe mas não um conjunto. A fim de evitar falar duma álgebra cujo domínio não é um conjunto poderíamos por exemplo assumir que os estados de todos os staas pertencem a um conjunto pré-determinado (v. exercício 1.2.7-3), mas feita esta ressalva não mais nos preocuparemos com este assunto.

Sistema nulo. Primeiro representaremos por *NIL* o staa $\langle\{\emptyset\},\emptyset,\emptyset\rangle$. Este é um caso particular de staa com um só estado (o estado inicial) e sem quaisquer transições. Designaremos este sistema por *nulo*.

Prefixação. A cada acção $\alpha \in Act$ faremos corresponder uma operação unária, representada por p_α, que a cada staa $\mathsf{S} = \langle P, T, \imath\rangle$ faz corresponder o staa

$$\mathsf{p}_\alpha(\mathsf{S}) \stackrel{\text{def}}{=} \langle P', T', \imath'\rangle \ ,$$

onde

$$\begin{aligned} P' &\stackrel{\text{def}}{=} \{\emptyset\} \cup \{\{x\} \mid x \in P\} \ , \\ \imath' &\stackrel{\text{def}}{=} \emptyset \ , \\ T' &\stackrel{\text{def}}{=} \{\langle\emptyset, \alpha, \{\imath\}\rangle\} \cup \{\langle\{x\}, \beta, \{y\}\rangle \mid \langle x, \beta, y\rangle \in T\} \ . \end{aligned}$$

Esta operação será designada por *prefixação (por α)*.

Composição paralela. Dados dois staas $\mathsf{S}_1 = \langle P_1, T_1, \imath_1\rangle$ e $\mathsf{S}_2 = \langle P_2, T_2, \imath_2\rangle$, a *composição paralela* (de S_1 e S_2) é o staa $\mathsf{par}(\mathsf{S}_1, \mathsf{S}_2) \stackrel{\text{def}}{=} \langle P, T, \imath\rangle$, onde

$$\begin{aligned} P &\stackrel{\text{def}}{=} P_1 \times P_2 \ , \\ \imath &\stackrel{\text{def}}{=} \langle \imath_1, \imath_2\rangle \ , \\ \langle x, y\rangle \stackrel{\alpha}{\to} \langle x', y'\rangle &\iff (x \stackrel{\alpha}{\to} x' \ \text{ e } \ y = y') \ \text{ ou } \ (x = x' \ \text{ e } \ y \stackrel{\alpha}{\to} y') \ . \end{aligned}$$

Escolha. Sejam agora $\mathsf{S}_1 = \langle P_1, T_1, \imath_1 \rangle$ e $\mathsf{S}_2 = \langle P_2, T_2, \imath_2 \rangle$ dois stas (não necessariamente staas). A *escolha* (entre S_1 e S_2) é $\mathsf{esc}(\mathsf{S}_1, \mathsf{S}_2) \stackrel{\text{def}}{=} \langle P, T, \imath \rangle$, onde

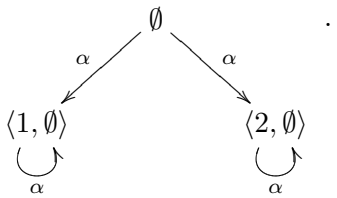
$$\begin{aligned} \imath &\stackrel{\text{def}}{=} \emptyset \,, \\ P &\stackrel{\text{def}}{=} \{\emptyset\} \cup (\{1\} \times P_1) \cup (\{2\} \times P_2) \,, \end{aligned}$$

e T é a menor relação de transição sobre Act tal que, para todo o $\alpha \in Act$ e $x, y \in P_i$,

$$\begin{aligned} \emptyset \xrightarrow{\alpha} \langle i, x \rangle \quad &\text{se} \quad \imath_i \xrightarrow{\alpha} x \text{ em } \mathsf{S}_i \,, \\ \langle i, x \rangle \xrightarrow{\alpha} \langle i, y \rangle \quad &\text{se} \quad x \xrightarrow{\alpha} y \text{ em } \mathsf{S}_i \,. \end{aligned}$$

Exemplo. 1. $\mathsf{esc}(NIL, NIL) = \langle \{\emptyset, \langle 1, \emptyset \rangle, \langle 2, \emptyset \rangle\}, \emptyset, \emptyset \rangle$.

2. Se $\mathsf{S} = \langle \{\emptyset\}, \{\langle \emptyset, \alpha, \emptyset \rangle\}, \emptyset \rangle$ então $\mathsf{esc}(\mathsf{S}, \mathsf{S})$ é o staa



.

O primeiro dos exemplos acima mostra que $\mathsf{esc}(\mathsf{S}_1, \mathsf{S}_2)$ pode não ser acessível, mesmo se S_1 e S_2 o forem. Isto acontecerá sse pelo menos um dos estados iniciais $\imath_1$ ou $\imath_2$ não for acessível a si próprio por meio dum traço diferente de ε. Para obviar a este problema definimos, dado um sta S, $\mathsf{staa}(\mathsf{S})$ como o staa cujos estados são exactamente os estados acessíveis de S (o que inclui o estado inicial) e cujas transições são exactamente aquelas das transições de S que partem e terminam em estados acessíveis.

Exercício. Formalize a definição de $\mathsf{staa}(\mathsf{S})$.

Definimos agora a *escolha* de S_1 e S_2, para staas, como sendo

$$\mathsf{esc}'(\mathsf{S}_1, \mathsf{S}_2) \stackrel{\text{def}}{=} \mathsf{staa}(\mathsf{esc}(\mathsf{S}_1, \mathsf{S}_2)) \,.$$

Teorema. *A relação de isomorfismo de staas é uma relação de congruência.*

1.2.2

A assinatura Proc tem como símbolos de operação as acções $\alpha \in Act$ (símbolos unários), e ainda os símbolos "$\mathbf{0}$" (constante), "+" (binário) e "$\|$" (binário).

A álgebra STAA é portanto uma Proc-álgebra:

$$\begin{aligned} \mathbf{0}_{\mathrm{STAA}} &= \mathit{NIL}\,, \\ \alpha_{\mathrm{STAA}} &= \mathsf{p}_\alpha\,, \\ +_{\mathrm{STAA}} &= \mathsf{esc}'\,, \\ \|_{\mathrm{STAA}} &= \mathsf{par}\,. \end{aligned}$$

Designaremos os termos de T_{Proc} por *termos de processo* e representá-los-emos genericamente por P, Q, P', P_1, etc.

Notação. Dados dois termos de processo P e Q, escreveremos usualmente $+PQ$ e $\|PQ$ na forma $P+Q$ e $P \,\|\, Q$, respectivamente, utilizando parênteses onde necessário, assumindo-se que a expressão $P+Q \,\|\, R$ significa $P+(Q \,\|\, R)$, isto é, representa o termo $+P\|QR$. Também assumiremos que $\alpha P \,\|\, Q$ se lê $(\alpha P) \,\|\, Q$ e não $\alpha(P \,\|\, Q)$. Também omitiremos usualmente $\mathbf{0}$ em termos da forma $\alpha\mathbf{0}$. Por exemplo, $\alpha+\beta\mathbf{0}\gamma\mathbf{0}$ será escrito como $\alpha(\beta+\gamma)$. Finalmente, por vezes utilizaremos um ponto nas prefixações, como no exemplo seguinte, onde se assume $\{\mathtt{inicio}, \mathtt{meio1}, \mathtt{fim1}, \mathtt{meio2}, \mathtt{fim2}\} \subseteq Act$:

$$\mathtt{inicio}.(\mathtt{meio1}.\mathtt{fim1}.\mathbf{0} + \mathtt{meio2}.\mathtt{fim2}.\mathbf{0})\,.$$

Por vezes escreveremos

- $\mathbf{0}$ em vez de NIL,
- $\alpha(\mathsf{S})$ em vez de $\mathsf{p}_\alpha(\mathsf{S})$,
- $(\mathsf{S} + \mathsf{T})$ em vez $\mathsf{esc}'(\mathsf{S}, \mathsf{T})$,
- $(\mathsf{S} \,\|\, \mathsf{T})$ em vez de $\mathsf{par}(\mathsf{S}, \mathsf{T})$,

Podendo também omitir parênteses nas prefixações, escrevendo, e.g., $\alpha\beta\gamma\mathsf{S}$ em vez de $\alpha(\beta(\gamma(\mathsf{S})))$.

Nota. Estas convenções correspondem a escrever termos em $T_{\mathsf{Proc}}\langle\{\text{“}\mathsf{S}\text{”}, \text{“}\mathsf{T}\text{”}, \ldots\}\rangle$.

1.2.3

Teorema. *A álgebra* STAA/$\cong$ *satisfaz as seguintes equações:*

1. $x + y = y + x$,
2. $x + (y + z) = (x + y) + z$,
3. $x \parallel y = y \parallel x$,
4. $x \parallel (y \parallel z) = (x \parallel y) \parallel z$,
5. $\mathbf{0} \parallel x = x$.

Exercício. Mostre, por meio de exemplos, que a equação

$$x + \mathbf{0} = x$$

não é satisfeita por $STAA/\cong$. [V. também o exercício 1.2.7-6.]

1.2.4

Sejam $s, t, u \in Act^*$. Diremos que s é uma *intercalação* de t e u se $s \lhd^t_u$, onde a relação ternária $\cdot \lhd^{\cdot}_{\cdot}$ é definida recursivamente:

$$\begin{array}{rcl} \varepsilon \lhd^t_u & \iff & t = u = \varepsilon \ , \\ \alpha s \lhd^t_u & \iff & \exists_{t'}(t = \alpha t' \ \text{ e } \ s \lhd^{t'}_u) \ \text{ ou } \ \exists_{u'}(u = \alpha u' \ \text{ e } \ s \lhd^t_{u'}) \ . \end{array}$$

Utilizaremos também a seguinte notação, para $t, u \in Act^*$ e $X, Y \subseteq Act^*$:

$$\begin{array}{rcl} t \otimes u & \stackrel{\text{def}}{=} & \{s \in Act^* \mid s \lhd^t_u\} \ , \\ X \otimes Y & \stackrel{\text{def}}{=} & \displaystyle\bigcup_{t \in X, u \in Y} t \otimes u \ , \end{array}$$

e escreveremos por vezes $s \in t \otimes u$ em vez de $s \lhd^t_u$.

Nota. Da definição resulta imediatamente a seguinte lei de distributividade:

$$X \otimes \bigcup_{i \in I} X_i = \bigcup_{i \in I} X \otimes X_i \ ,$$

onde I é um conjunto de indexação qualquer.

Lema. *Sejam* $\mathsf{S} = \langle P, T, \imath\rangle$ *e* $\mathsf{T} = \langle Q, U, \jmath\rangle$ *staas. Então, em* par(S,T) *tem-se, para* $x, z \in P$, $y, w \in Q$ *e* $s \in Act^*$,

$$\langle x, y\rangle \xrightarrow{s} \langle z, w\rangle \iff \exists_{t,u}(s \lhd^t_u \quad e \quad x \xrightarrow{t} z \quad e \quad y \xrightarrow{u} w) \ .$$

Prova. Por indução no comprimento de s. A base da indução é dada por $s = \varepsilon$ e é imediata. Seja agora $s = \alpha s'$. Tem-se

$$\begin{array}{rcl}
\langle x, y\rangle \xrightarrow{s} \langle x', y'\rangle & \iff & \exists_{z,w}(\langle x, y\rangle \xrightarrow{\alpha} \langle z, w\rangle \xrightarrow{s'} \langle x', y'\rangle) \\
& \iff & \exists_z(x \xrightarrow{\alpha} z \ \text{ e } \ \langle z, y\rangle \xrightarrow{s'} \langle x', y'\rangle) \\
& & \ \text{ ou } \ \exists_w(y \xrightarrow{\alpha} w \ \text{ e } \ \langle x, w\rangle \xrightarrow{s'} \langle x', y'\rangle) \\
& \iff & \exists_z(x \xrightarrow{\alpha} z \ \text{ e } \ \exists_{t',u}(z \xrightarrow{t'} x' \ \text{ e } \ y \xrightarrow{u} y' \ \text{ e } \ s' \lhd^{t'}_u)) \\
& & \ \text{ ou } \ \exists_w(y \xrightarrow{\alpha} w \ \text{ e } \ \exists_{t,u'}(w \xrightarrow{u'} y' \ \text{ e } \ x \xrightarrow{t} x' \ \text{ e } \ s' \lhd^t_{u'})) \\
& & \text{(Hip. de indução)} \\
& \iff & \exists_{t',u}(x \xrightarrow{\alpha t'} x' \ \text{ e } \ u \xrightarrow{u} y' \ \text{ e } \ s' \lhd^{t'}_u)) \\
& & \ \text{ ou } \ \exists_{t,u'}(x \xrightarrow{t} x' \ \text{ e } \ y \xrightarrow{\alpha u'} y' \ \text{ e } \ s' \lhd^t_{u'})) \\
& \iff & \exists_{t,u}(x \xrightarrow{t} x' \ \text{ e } \ y \xrightarrow{u} y' \ \text{ e } \ \exists_{t'}(t = \alpha t' \ \text{ e } \ s' \lhd^{t'}_u)) \\
& & \ \text{ ou } \ \exists_{t,u}(x \xrightarrow{t} x' \ \text{ e } \ y \xrightarrow{u} y' \ \text{ e } \ \exists_{u'}(u = \alpha u' \ \text{ e } \ s' \lhd^t_{u'})) \\
& \iff & \exists_{t,u}[x \xrightarrow{t} x' \ \text{ e } \ y \xrightarrow{u} y' \ \text{ e } \\
& & (\exists_{t'}(t = \alpha t' \ \text{ e } \ s' \lhd^{t'}_u) \ \text{ ou } \ \exists_{u'}(u = \alpha u' \ \text{ e } \ s' \lhd^t_{u'}))] \\
& \iff & \exists_{t,u}(x \xrightarrow{t} x' \ \text{ e } \ y \xrightarrow{u} y' \ \text{ e } \ s \lhd^t_u) \\
& & \text{(Por def. de } s \lhd^t_u) \ \blacksquare
\end{array}$$

1.2.5

Não é muito lícito considerar STAA uma álgebra de processos pois um staa representa uma máquina mas um processo é apenas o comportamento *observável* duma máquina. Como vimos, existem várias noções de processo, dependendo das noções de observação em causa. Vamos agora analisar o modelo de traços **T** (v. exemplo 1.1.7) do ponto de vista algébrico. Um modo de equipar **T** com uma estrutura de Proc-álgebra é fazer:

1. $\mathbf{0_T} = \{\varepsilon\}$,
2. $+_{\mathbf{T}} = \cup$,
3. $\|_{\mathbf{T}} = \otimes$,
4. $\alpha_{\mathbf{T}} = \lambda X.\{\varepsilon\} \cup \alpha X$,

onde $\alpha X \stackrel{\text{def}}{=} \{\alpha t \mid t \in X\}$.

Para justificar que estas operações de facto tornam $\mathbf{T}$ uma Proc-álgebra deveríamos provar que $X \otimes Y$ é não vazio e fechado para prefixos se X e Y o forem (os outros casos são simples)—i.e., mostrar que as operações sobre conjuntos de traços atrás definidas são *fechadas* para o subconjunto $\mathbf{T}$. Contudo, não faremos isso directamente, pois o resultado será corolário do seguinte:

Proposição. *Sejam* S *e* T *staas. Então,*

1. $\mathcal{T}(NIL) = \{\varepsilon\}$,
2. $\mathcal{T}(\mathsf{p}_\alpha(\mathsf{S})) = \{\varepsilon\} \cup \alpha\mathcal{T}(\mathsf{S})$,
3. $\mathcal{T}(\mathsf{esc}'(\mathsf{S},\mathsf{T})) = \mathcal{T}(\mathsf{S}) \cup \mathcal{T}(\mathsf{T})$,
4. $\mathcal{T}(\mathsf{par}(\mathsf{S},\mathsf{T})) = \mathcal{T}(\mathsf{S}) \otimes \mathcal{T}(\mathsf{T})$.

Por outras palavras, $\mathcal{T} : \mathrm{STAA} \to \mathbf{T}$ é um homomorfismo de Proc-álgebras.

Prova. Os primeiros três casos são simples de verificar e o quarto é corolário do Lema 1.2.4. ∎

Corolário. *Sejam* $X, Y \subseteq Act^*$ *não vazios e fechados para prefixos. Então* $X \otimes Y$ *também o é.*

Prova. Pelo Exemplo (e exercício 1.1.8-8) resulta que existem staas S e T tais que $X = \mathcal{T}(\mathsf{S})$ e $Y = \mathcal{T}(\mathsf{T})$. Logo, pelo teorema anterior existe um staa, $\mathsf{par}(\mathsf{S},\mathsf{T})$, cujos traços são exactamente os do conjunto $X \otimes Y$; logo, novamente pelo Exemplo , $X \otimes Y$ é não vazio e fechado para prefixos. ∎

1.2.6

Lema. *Sejam* X, Y *e* Z *conjuntos não vazios de traços, e fechados para prefixos. Então,*

$$X \otimes (Y \otimes Z) = (X \otimes Y) \otimes Z \ .$$

Prova. Sendo os conjuntos não vazios e fechados para prefixos, existem staas S, T e U tais que $X = \mathcal{T}(\mathsf{S})$, $Y = \mathcal{T}(\mathsf{T})$ e $Z = \mathcal{T}(\mathsf{U})$. Tem-se então:

$$\begin{array}{rcll} X \otimes (Y \otimes Z) & = & \mathcal{T}(\mathsf{S} \parallel (\mathsf{T} \parallel \mathsf{U})) & (\mathcal{T} \text{ é homomorfismo.}) \\ & = & \mathcal{T}((\mathsf{S} \parallel \mathsf{T}) \parallel \mathsf{U}) & (\text{Pelo Teorema 1.2.3, e } \cong \subseteq \sim_{\mathcal{T}}.) \\ & = & (X \otimes Y) \otimes Z & (\mathcal{T} \text{ é homomorfismo.}) \ ∎ \end{array}$$

Teorema. *A álgebra* $\mathbf{T}$ *satisfaz as seguintes equações:*

1. $x + (y + z) = (x + y) + z$,
2. $x + \mathbf{0} = x$,
3. $x + y = y + x$,
4. $x + x = x$,
5. $x \parallel (y \parallel z) = (x \parallel y) \parallel z$,
6. $x \parallel \mathbf{0} = x$,
7. $x \parallel y = y \parallel x$,
8. $\alpha(x + y) = \alpha x + \alpha y$,
9. $\alpha x \parallel \beta y = \alpha(x \parallel \beta y) + \beta(\alpha x \parallel y)$,
10. $(x + y) \parallel z = x \parallel z + y \parallel z$.

Prova. 1-4: imediato, pelas propriedades da união de conjuntos. 5: pelo lema. 6,7: simples (pode provar-se via STAA, como no lema, ou directamente—v. exercício 1.2.7-7). 8: resulta da distributividade da concatenação de linguagens. 9: tem-se

$$\begin{aligned}(\{\varepsilon\} \cup \alpha X) \otimes (\{\varepsilon\} \cup \beta Y) &= \{\varepsilon\} \cup \{\varepsilon\} \otimes \beta Y \cup \alpha X \otimes \{\varepsilon\} \cup \alpha X \otimes \beta Y \\ &= \{\varepsilon\} \cup \alpha X \cup \beta Y \cup \alpha(X \otimes \beta Y) \cup \beta(\alpha X \otimes Y) \\ &\quad \text{(Pelo exercício 1.2.7-7.)} \\ &= \{\varepsilon\} \cup \alpha(X \otimes (\{\varepsilon\} \cup \beta Y)) \cup \beta((\{\varepsilon\} \cup \alpha X) \otimes Y)\ .\end{aligned}$$

10: resulta da distributividade de $\otimes$ sobre a união de conjuntos. ∎

1.2.7 Exercícios

1. Prove que em $\mathsf{p}_\alpha(\mathsf{S})$ se tem, dado qualquer $t \in Act^*$, $\{x\} \xrightarrow{t} \{y\}$ sse $x \xrightarrow{t} y$ em S.
2. Sejam $\mathsf{S} = \langle P, T, \imath\rangle$ e $\mathsf{T} = \langle Q, U, \jmath\rangle$. Prove que em $\mathsf{esc}'(\mathsf{S}, \mathsf{T})$ se tem, dado qualquer $t \in Act^*$,
   (a) $\langle 1, x\rangle \xrightarrow{t} \langle 1, y\rangle$ sse $x \xrightarrow{t} y$ em S,
   (b) se $t \neq \varepsilon$ então $\emptyset \xrightarrow{t} \langle 1, x\rangle \iff \imath \xrightarrow{t} x$,

(c) $\emptyset \xrightarrow{t} z$ sse uma das seguintes condições se verificar:

- $z = \emptyset$ e $t = \varepsilon$;
- ou $z = \langle 1, x\rangle$ para algum $x \in P$ tal que $\imath \xrightarrow{t} x$ em S;
- ou $z = \langle 2, y\rangle$ para algum $y \in Q$ tal que $\jmath \xrightarrow{t} y$ em T.

3. Seja $\mathcal{E}$ um conjunto, e defina-se staa do modo habitual, mas com a restrição de que o conjunto de estados deve ser um subconjunto de $\mathcal{E}$. De acordo com esta definição, a classe STAA de todos os staas será um conjunto. Que propriedades deverá $\mathcal{E}$ ter para que seja possível definir as operações *NIL*, p_α, esc' e par sobre STAA? Dê um exemplo dum tal conjunto $\mathcal{E}$ que seja contável.

4. Considere a seguinte definição alternativa da operação de prefixação por α:

$$\mathsf{p}_\alpha(\langle P, T, \imath\rangle) = \langle P \cup \{P\}, T \cup \{\langle P, \alpha, \imath\rangle\}, P\rangle \ .$$

Justifique que o novo estado inicial é satisfatório. Analise esta solução em termos do exercício anterior: ainda será possível existir um conjunto $\mathcal{E}$ tal que STAA seja um conjunto fechado para p_α?

5. Seja Est a assinatura algébrica com os símbolos "$\bullet$"(constante) e "."(binário). Seja $\mathrm{STAA}_{\mathsf{Est}}$ a classe dos staas cujos estados são termos de T_{Est}.

   (a) Justifique que a classe $\mathrm{STAA}_{\mathsf{Est}}$ é um conjunto.

   (b) Mostre que é possível nesta classe definir operações análogas a *NIL*, p_α, esc' e par. [Sugestão: utilize $\bullet$ em vez do conjunto vazio, represente o conjunto singular $\{x\}$ por $.\bullet x$, o par $\langle x, y\rangle$ por $.xy$, $\langle 1, x\rangle$ por $.\bullet x$ e $\langle 2, x\rangle$ por $.\bullet.\bullet x$.]

   (c) O conjunto T_{Est} é contável. Diga como pode descrever, ainda com base na assinatura Est, um conjunto de estados não contável e que permita definir as operações acima.

6. Mostre que os staas acíclicos formam uma subálgebra de STAA, e designe-a por STAAA. Justifique que a relação de isomorfismo é uma relação de congruência sobre STAAA e mostre que a álgebra $\mathrm{STAAA}/\cong$ satisfaz a equação

$$x + \mathbf{0} = x \ .$$

7. Sejam $s, t, u \in Act^*$ e $X, Y \subseteq Act^*$. Prove que:

   (a) $s \lhd_\varepsilon^t \iff s = t$;

(b) $s \lhd^t_u \iff s \lhd^u_t$;

(c) $s \lhd^{\alpha t}_{\beta u} \iff \exists_{s'}((s = \alpha s' \text{ e } s' \lhd^{t'}_{\beta u}) \vee (s = \beta s' \text{ e } s' \lhd^{\alpha t}_{u'}))$;

(d) $t \otimes \varepsilon = t$;

(e) $t \otimes u = u \otimes t$;

(f) $\alpha t \otimes \beta u = \alpha(t \otimes \beta u) \cup \beta(\alpha t \otimes u)$;

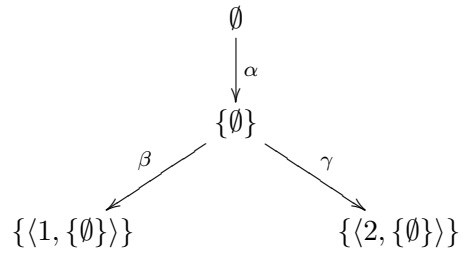
(g) $X \otimes \{\varepsilon\} = X$;

(h) $X \otimes Y = Y \otimes X$;

(i) $\alpha X \otimes \beta Y = \alpha(X \otimes \beta Y) \cup \beta(\alpha X \otimes Y)$.

1.3 Linguagens de descrição de processos

Dada a estrutura algébrica dos processos, é natural descrevê-los por meio de expressões algébricas apropriadas. Por exemplo, vimos na secção anterior que podemos especificar o staa



por meio da expressão $\mathsf{p}_\alpha(\mathsf{esc}'(\mathsf{p}_\beta(NIL), \mathsf{p}_\gamma(NIL)))$, ou, equivalentemente, recorrendo à assinatura Proc, pela expressão $\alpha(\beta + \gamma)$, que é um termo de T_{Proc}. Como vimos, esta expressão denota também o conjunto de traços $\{\varepsilon, \alpha, \alpha\beta, \alpha\gamma\}$. Nesta secção iremos utilizar linguagens baseadas na assinatura Proc para descrever processos.

1.3.1

Seja $\mathsf{S} = \langle P, T, \imath \rangle$ um staa. Um estado $x \in P$ diz-se *cíclico* se existe algum traço t de comprimento não nulo tal que $x \xrightarrow{t} x$. S diz-se *cíclico* se tem algum estado cíclico, e *acíclico* em caso contrário. Diremos também que S é *finito* se for simultaneamente acíclico e tiver um número finito de estados.

O comportamento dum staa finito é finito, no seguinte sentido:

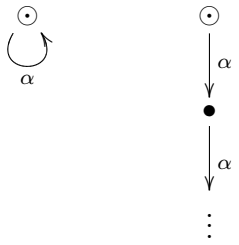
Proposição. *Seja* S *um staa finito. Então existe* $n \geq 0$ *tal que qualquer traço* $t \in \mathcal{T}(\mathsf{S})$ *tem comprimento majorado por* n.

Prova. Seja N o número de estados de S e assuma-se que existe um traço $\alpha_1 \ldots \alpha_N$ em $\mathcal{T}(\mathsf{S})$. Então existem estados $x_0, \ldots, x_N$, com $x_0 = \imath$, tais que $x_0 \stackrel{\alpha_1}{\to} x_1 \stackrel{\alpha_2}{\to} \ldots \stackrel{\alpha_N}{\to} x_N$. Como só há N estados, terá de ter-se $x_i = x_j$ para algum par $i \neq j$, e portanto S é cíclico, o que é uma contradição. Logo, o comprimento máximo dos traços de S é $n \leq N - 1$. ∎

Teorema. *Seja P um termo de* Proc. *Então P_{STAA} é finito.*

Prova. Por indução na estrutura dos termos. ∎

Deste teorema resulta que com os termos de T_{Proc} apenas podemos especificar staas cujo comportamento é finito, o que exclui staas tão simples como os seguintes:



1.3.2

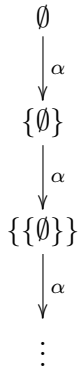
Para especificar staas não finitos utilizaremos geradores e relações (v. apêndice). Uma *especificação de processo* sobre um conjunto de geradores G é um par $\langle P, \rho \rangle$, onde $P \in T_{\mathsf{Proc}}\langle G \rangle$ e ρ é um conjunto de relações em $T_{\mathsf{Proc}}\langle G \rangle$.

Notação. Utilizaremos frequentemente $P, P', Q, Q_1, \ldots$ para designar termos arbitrários em $T_{\mathsf{Proc}}\langle G \rangle$, reservando $A, B, B', C \ldots$ para os geradores.

São especificações de processo, por exemplo:

- $\langle A, \{A = \alpha A\} \rangle$,
- $\langle \beta A + \gamma, \{\alpha A = A\} \rangle$,
- $\langle \alpha A \parallel B, \{A = B, B = \alpha + \beta A\} \rangle$,
- $\langle A, \{\alpha A = \beta A\} \rangle$,
- $\langle A, \emptyset \rangle$.

Algumas destas especificações têm "soluções"únicas; por exemplo, a única maneira de respeitar a relação $A = \alpha A$ em STAA é atribuir ao gerador A o staa



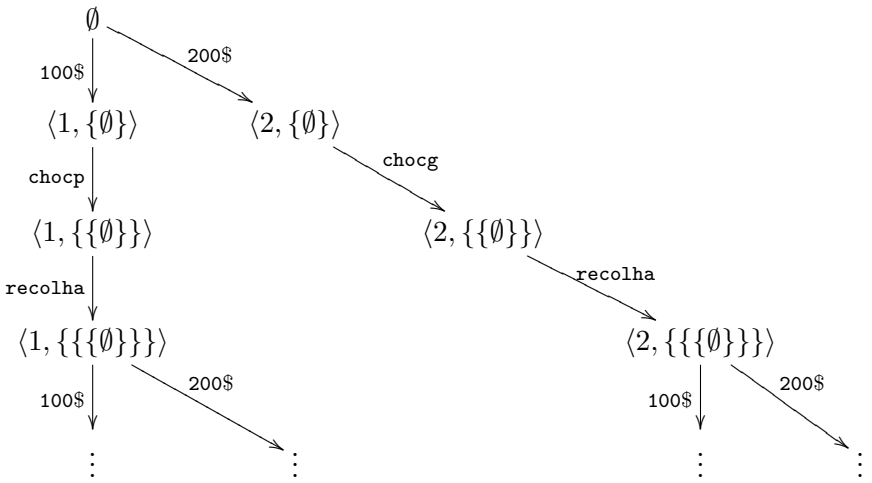
(v. exercício 1.3.3-8), ou, em **T**, o conjunto de traços $\{\alpha\}^*$. Já a especificação $\langle P, \{P = P\}\rangle$ tem múltiplas soluções, enquanto que $\langle P, \{\alpha P = \beta Q\}\rangle$ não tem nenhuma, nem em STAA nem em **T** (verifique).

Exemplo. Seja $V \in G$ um gerador e ρ o conjunto singular com a sequinte relação:

$$V = \texttt{100\$.chocp.recolha}.V + \texttt{200\$.chocg.recolha}.V \ .$$

A especificação $\langle V, \rho\rangle$ representa uma máquina de venda de chocolates cujo comportamento é o seguinte: o utilizador pode inserir uma moeda de 100\$ (acção `100$`) ou de 200\$ (acção `200$`), após o que a máquina faz sair um chocolate pequeno (acção `chocp`) ou grande (acção `chocg`), respectivamente; nesse momento o utilizador recolhe o chocolate (acção `recolha`), após o que

a máquina regressa à condição inicial. O staa denotado pela especificação é



Exercício. Obtenha um staa cíclico, com um número finito de estados, e com os mesmos traços de menu ($\mathcal{RT}$) do staa acima.

1.3.3 Exercícios

1. Assuma $Act = \{a, b, c, d, e, f\}$. Represente graficamente os staas denotados pelos seguintes termos de processo.

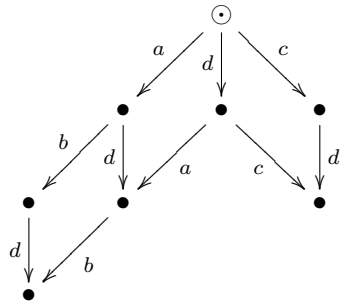
   (a) ab
   (b) $ab + ba$
   (c) $a \parallel b$
   (d) $ab \parallel c$
   (e) $a(b \parallel c)$
   (f) $(cab + acb) + abc$
   (g) $cab + a(b \parallel c)$
   (h) $ab \parallel cd$
   (i) $ab + ac$
   (j) $a(b + c)$
   (k) $a(b + cd) + a(ce + f)$
   (l) $a(b + ce) + a(cd + f)$
   (m) $(a + b) \parallel c$

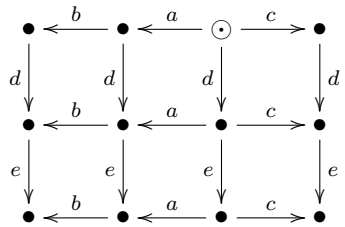
(n) $(a + b) + c$

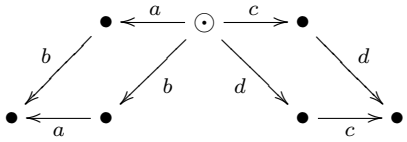
(o) $a + (b + c)$

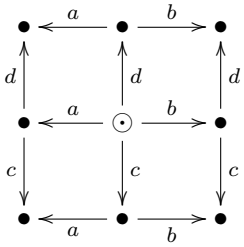
(p) $(a \parallel b) \parallel c$

2. Escreva termos de processo que denotem staas isomorfos aos seguintes:









3. Represente graficamente o staa denotado pelo termo $((a + b) \parallel (c + d)) \parallel e$.

4. Mostre que nenhum termo de processo denota um staa isomorfo a $\odot \overset{\alpha}{\underset{\beta}{\rightrightarrows}} \bullet$.

5. Modifique a especificação da máquina de venda de chocolates de modo a que após inserir 100$ seja possível inserir mais 100$ para comprar um chocolate grande.

6. Modifique a especificação da máquina de venda de chocolates de modo a que após ter inserido 200$ seja possível comprar dois chocolates pequenos em vez de um grande.

7. Utilizando as mesmas acções (`100$`, `200$`, `chocp`, `chocg` e `recolha`), especifique uma máquina de venda de chocolates W que obedeça às seguintes restrições:

   (a) W não obtém lucros nem perdas;

   (b) W não pode acumular um crédito de mais de 400$ (o que pode impedir a colocação duma moeda);

   (c) não há espaço para mais do que um chocolate à saída da máquina (o que pode bloquear os botões `chocp` e `chocg`).

8. Obtenha soluções em STAA, e mostre que são únicas, para as seguintes especificações de processo:

   (a) $\langle A, \{A = \alpha A\}\rangle$;

   (b) $\langle A, \{A = \alpha A + \beta A\}\rangle$;

   (c) $\langle A, \{A = \alpha\beta A + \gamma A\}\rangle$;

   (d) $\langle A, \{A = \alpha(A \parallel \beta)\}\rangle$.

9. Para as três primeiras alíneas do exercício anterior obtenha expressões regulares que denotem o conjunto de traços do sistema especificado.

10. Seja $Act = \{\texttt{inc}, \texttt{dec}\}$. Considere a especificação $\langle C_0, \rho\rangle$ dum *contador*, onde ρ é o conjunto (numerável) de relações descrito a seguir:

$$\begin{aligned} C_0 &= \texttt{inc}.C_1 \\ C_{n+1} &= \texttt{inc}.C_{n+2} + \texttt{dec}.C_n \quad (n \geq 0) \end{aligned}$$

    (a) Obtenha uma solução desta especificação em STAA.

    (b) Uma especificação alternativa (e finita) dum contador pode ser dada por $\langle C, \{C = \texttt{inc}.(C \parallel \texttt{dec})\}\rangle$. Com base no resultado do exercício 8d justifique que STAA não é um bom modelo de processos.

(c) Mostre que ambas as especificações denotam staas com os mesmos traços.

11. Especifique:

    (a) uma pilha de booleanos;

    (b) uma fila de espera de booleanos;

    (c) um saco de booleanos, sem utilizar $\|$;

    (d) um saco de booleanos, com apenas uma relação.

1.4 Bissimulação

Nesta secção estudaremos uma noção de equivalência de staas que, ao contrário das anteriores, não é baseada numa noção explícita de "observação"(e.g., traços, traços de falha, etc). Esta nova equivalência, usualmente conhecida por *bissimulação*, foi introduzida independentemente por Park e por Milner.

1.4.1

Sejam $\mathsf{S} = \langle P, T, \imath \rangle$ e $\mathsf{T} = \langle Q, U, \jmath \rangle$ dois staas. O nosso primeiro objectivo é definir uma relação $\sim \subseteq P \times Q$ com o seguinte significado: $x \sim y$ se o sistema S tem, no estado $x \in P$, o mesmo comportamento observável que o sistema T no estado $y \in Q$. Para tal adoptaremos a seguinte "definição"circular:

$$x \sim y \iff \begin{cases} \forall_{x' \in P}(x \xrightarrow{\alpha} x' \Rightarrow \exists_{y' \in Q}(y \xrightarrow{\alpha} y' \text{ e } x' \sim y')) \,, e \\ \forall_{y' \in Q}(y \xrightarrow{\alpha} y' \Rightarrow \exists_{x' \in P}(x \xrightarrow{\alpha} x' \text{ e } x' \sim y')) \,. \end{cases}$$

O significado intuitivo destas condições é o de que dois estados x e y são equivalentes se conseguem imitar-se um ao outro: se x executa α, atingindo um estado x', então y deve ser capaz de executar α, e além disso atingir um estado y' equivalente a x'; e o mesmo se é y que começa a executar (podemos pensar nesta noção como uma espécie de jogo).

Contudo, esta condição não constitui uma definição de $\sim$, porque em geral há várias relações que a satisfazem. Uma solução é considerar que x_0 e y_0 são equivalentes, escrevendo $x_0 \sim y_0$, se existe alguma relação $R \subseteq P \times Q$ com as propriedades acima, i.e., tal que, para quaisquer $x \in P$ e $y \in Q$,

$$xRy \iff \begin{cases} \forall_{x' \in P}(x \xrightarrow{\alpha} x' \Rightarrow \exists_{y' \in Q}(y \xrightarrow{\alpha} y' \text{ e } x'Ry')) \,, e \\ \forall_{y' \in Q}(y \xrightarrow{\alpha} y' \Rightarrow \exists_{x' \in P}(x \xrightarrow{\alpha} x' \text{ e } x'Ry')) \,, \end{cases}$$

e tal que $x_0 R y_0$. Diremos que uma tal relação R é *de imitação*. Portanto estamos a definir:

$$x \sim y \overset{\text{def}}{\iff} xRy \text{ para alguma relação de imitação } R \, ;$$

ou, equivalentemente,

$$\sim \overset{\text{def}}{=} \bigcup \{ R \subseteq P \times Q \mid R \text{ é de imitação } \} \, .$$

Uma forma elegante de resumir estes conceitos é o seguinte: defina-se uma função $\phi : 2^{P \times Q} \to 2^{P \times Q}$ tal que

$$\begin{array}{rcl} \phi(R) \overset{\text{def}}{=} \{ \langle x, y \rangle \in P \times Q & \mid & \forall_{x' \in P} (x \xrightarrow{\alpha} x' \Rightarrow \exists_{y' \in Q} (y \xrightarrow{\alpha} y' \text{ e } x' R y')) \\ & \text{e} & \forall_{y' \in Q} (y \xrightarrow{\alpha} y' \Rightarrow \exists_{x' \in P} (x \xrightarrow{\alpha} x' \text{ e } x' R y')) \} \, ; \end{array}$$

A condição de que R é de imitação pode agora ser simplesmente expressa pela equação $R = \phi(R)$; isto é, as relações de imitação são exactamente os pontos fixos da função ϕ.

Proposição. *A função ϕ é monótona.*

Prova. Se $R \subseteq R'$ então $\langle x, y \rangle \in \phi(R)$ significa que

$$\begin{array}{l} \forall_{x' \in P} (x \xrightarrow{\alpha} x' \Rightarrow \exists_{y' \in Q} (y \xrightarrow{\alpha} y' \text{ e } x' R y')) \, , e \\ \forall_{y' \in Q} (y \xrightarrow{\alpha} y' \Rightarrow \exists_{x' \in P} (x \xrightarrow{\alpha} x' \text{ e } x' R y')) \, . \end{array}$$

Se $R \subseteq R'$ tem-se que $x' R' y'$ sempre que $x' R y'$, e portanto também se tem

$$\begin{array}{l} \forall_{x' \in P} (x \xrightarrow{\alpha} x' \Rightarrow \exists_{y' \in Q} (y \xrightarrow{\alpha} y' \text{ e } x' R' y')) \, , e \\ \forall_{y' \in Q} (y \xrightarrow{\alpha} y' \Rightarrow \exists_{x' \in P} (x \xrightarrow{\alpha} x' \text{ e } x' R' y')) \, , \end{array}$$

isto é, $\langle x, y \rangle \in \phi(R')$. ∎

Da teoria dos reticulados (v. [3, 4]) resulta imediatamente que ϕ tem um ponto fixo, e que existe o maior de todos os seus pontos fixos; este é explicitamente dado por $\bigcup \{ R \subseteq P \times Q \mid R = \phi(R) \}$, ou seja, coincide precisamente com a relação $\sim$, a qual é portanto também uma relação de imitação, e em particular a maior de todas elas.

1.4.2

Os resultados acima mostram que para estabelecer $x \sim y$ basta encontrar uma relação de imitação R tal que xRy. Mas há ainda um modo mais simples: da teoria dos reticulados resulta também que o maior ponto fixo duma função monótona coincide com o maior pré-ponto-fixo; neste caso isso significa que a relação $\sim$ pode também ser dada por

$$\sim = \bigcup \{R \subseteq P \times Q \mid R \subseteq \phi(R)\} \ .$$

A um pré-ponto-fixo de ϕ, i.e., uma relação tal que $R \subseteq \phi(R)$, chamaremos uma *relação de bissimulação*, ou simplesmente uma *bissimulação*. Por outras palavras, uma bissimulação é uma relação R tal que

$$xRy \Rightarrow \begin{cases} \forall_{x' \in P}(x \xrightarrow{\alpha} x' \Rightarrow \exists_{y' \in Q}(y \xrightarrow{\alpha} y' \ \text{ e } \ x'Ry')) \ , e \\ \forall_{y' \in Q}(y \xrightarrow{\alpha} y' \Rightarrow \exists_{x' \in P}(x \xrightarrow{\alpha} x' \ \text{ e } \ x'Ry')) \ , \end{cases}$$

e tem-se

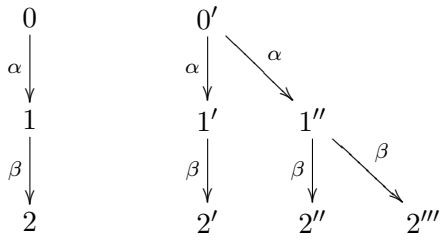
Teorema. *$x \sim y$ sse existe uma bissimulação R tal que xRy.*

Dados staas $\mathsf{S} = \langle P, T, \imath \rangle$ e $\mathsf{T} = \langle Q, U, \jmath \rangle$ diremos que uma bissimulação $R \subseteq P \times Q$ é uma *bissimulação entre* S e T. A uma bissimulação entre S e S chamaremos também uma *bissimulação sobre* S, ou *em* S.

1.4.3

Sejam S e T staas. Diremos que $\mathsf{S} = \langle P, T, \imath \rangle$ e $\mathsf{T} = \langle Q, U, \jmath \rangle$ são *bissimilares*, ou *bissimuláveis*, se $\imath \sim \jmath$.

Exemplo. Os seguintes staas, com estados iniciais 0 e $0'$, são bissimilares:



A relação

$$\{\langle 0, 0' \rangle, \langle 1, 1' \rangle, \langle 1, 1'' \rangle, \langle 2, 2' \rangle, \langle 2, 2'' \rangle, \langle 2, 2''' \rangle\}$$

é uma bissimulação.

Teorema. *Sejam* $\mathsf{S}_1 = \langle P_1, T_1, \imath_1 \rangle$, $\mathsf{S}_2 = \langle P_2, T_2, \imath_2 \rangle$ *e* $\mathsf{S}_3 = \langle P_3, T_3, \imath_3 \rangle$ *staas.*

1. Δ_{P_1} *é uma bissimulação sobre* S_1.
2. *Se* $R \subseteq P_1 \times P_2$ *é uma bissimulação entre* S_1 *e* S_2 *então a relação*

$$R^{-1} = \{\langle y, x \rangle \mid xRy\}$$

   é uma bissimulação entre S_2 *e* S_1.
3. *Se* $R \subseteq P_1 \times P_2$ *é uma bissimulação entre* S_1 *e* S_2 *e* $S \subseteq P_2 \times P_3$ *é uma bissimulação entre* S_2 *e* S_3 *então* $S \circ R$ *é uma bissimulação entre* S_1 *e* S_3.
4. *Se* $\{R_i\}_{i \in I}$ *é uma família de bissimulações (indexada por um conjunto* I *arbitrário) entre* S_1 *e* S_2*, então também o é a relação* $\bigcup_{i \in I} R_i$.

1.4.4

Sejam $\mathsf{S} = \langle P, T, \imath \rangle$ e $\mathsf{T} = \langle Q, U, \jmath \rangle$ staas. Dizemos que uma relação $R \subseteq P \times Q$ é uma *bissimulação a menos de* $\sim$ se

$$xRy \Rightarrow \begin{cases} \forall_{x' \in P}(x \xrightarrow{\alpha} x' \Rightarrow \exists_{x'' \in P, y', y'' \in Q}(y \xrightarrow{\alpha} y' \text{ e } x' \sim x'' R y'' \sim y')) \,, e \\ \forall_{y' \in Q}(y \xrightarrow{\alpha} y' \Rightarrow \exists_{x', x'' \in P, y'' \in Q}(x \xrightarrow{\alpha} x' \text{ e } x' \sim x'' R y'' \sim y')) \,, \end{cases}$$

Por outras palavras, R é uma bissimulação a menos de $\sim$ sse

$$R \subseteq \phi(\sim_{\mathsf{T}} \circ R \circ \sim_{\mathsf{S}}) \,,$$

onde $\sim_{\mathsf{S}}$ e $\sim_{\mathsf{T}}$ representam as relações de bissimilaridade sobre S e T, respectivamente.

Teorema. *Sejam* $\mathsf{S} = \langle P, T, \imath \rangle$ *e* $\mathsf{T} = \langle Q, U, \jmath \rangle$ *staas, e seja* R *uma bissimulação a menos de* $\sim$ *entre* S *e* T*. Então:*

1. $\sim_{\mathsf{T}} \circ R \circ \sim_{\mathsf{S}}$ *é uma bissimulação entre* S *e* T.
2. $xRy \Rightarrow x \sim y$.

Prova. É simples ver que a operação $j : 2^{P \times Q} \to 2^{P \times Q}$ definida por $j(S) = \sim_{\mathsf{T}} \circ S \circ \sim_{\mathsf{S}}$ é um operador de fecho em $2^{P \times Q}$. Além disso o operador

ϕ preserva os elementos fechados, i.e., $j(\phi(S)) = \phi(S)$ se $j(S) = S$ (v. exercício 1.4.13-3). Daqui resulta que $\phi(j(R))$ é fechado e que portanto

$$R \subseteq \phi(j(R)) \iff j(R) \subseteq \phi(j(R)) \ .$$

Logo, $j(R)$ é uma bissimulação porque R é uma bissimulação a menos de $\sim$. Sejam agora x e y estados tais que xRy. Tem-se então $\langle x, y\rangle \in j(R)$ por uma das propriedades dos operadores de fecho, e portanto x e y são bissimilares. ∎

1.4.5

Dado um staa $\mathsf{S} = \langle P, T, \imath\rangle$ e um estado $x \in P$, definimos

$$\mathcal{RT}(x) \stackrel{\text{def}}{=} \mathcal{RT}(\mathsf{staa}(\langle P, T, x\rangle)) \ ;$$

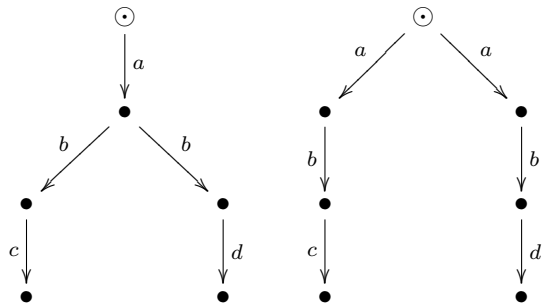
$\mathcal{RT}(x)$ é portanto o conjunto de traços de menu observáveis a partir do estado x.

Lema. *Sejam $\mathsf{S} = \langle P, T, \imath\rangle$ e $\mathsf{T} = \langle Q, U, \jmath\rangle$ staas, $x \in P$ e $y \in Q$. Se $x \sim y$ então $\mathcal{RT}(x) = \mathcal{RT}(y)$.*

Prova. (Esboço.) Seja $x \sim y$ e $\langle X_0, \alpha_1, \ldots, X_n\rangle \in \mathcal{RT}(x)$. Prova-se que $\langle X_0, \alpha_1, \ldots, X_n\rangle \in \mathcal{RT}(y)$, por indução em n, donde resulta $\mathcal{RT}(x) \subseteq \mathcal{RT}(y)$. Por simetria conclui-se que $\mathcal{RT}(y) \subseteq \mathcal{RT}(x)$. ∎

Teorema. $\sim \subsetneq \sim_{\mathcal{RT}}$.

Prova. Sejam $\mathsf{S} = \langle P, T, \imath\rangle$ e $\mathsf{T} = \langle Q, U, \jmath\rangle$. Tem-se $\mathsf{S} \sim_{\mathcal{RT}} \mathsf{T}$ sse $\mathcal{RT}(\imath) = \mathcal{RT}(\jmath)$, e portanto, pelo Lema, $\mathsf{S} \sim_{\mathcal{RT}} \mathsf{T}$ se $\mathsf{S} \sim \mathsf{T}$. O facto de que $\sim \neq \sim_{\mathcal{RT}}$ resulta do contra-exemplo seguinte:



Estes dois staas têm os mesmos traços de menu mas não são bissimilares (verifique). ∎

1.4.6

Teorema. *A relação de bissimilaridade entre staas,* $\sim$*, é uma relação de congruência sobre* STAA*.*

Aos elementos da álgebra STAA/$\sim$ chamamos staas *módulo bissimulação*, ou staas *a menos de bissimulação*. Nas situações em que a relação de equivalência comportamental de staas é a bissimilaridade os processos são justamente os staas a menos de bissimulação. O quociente STAA/$\sim$ é portanto o nosso primeiro modelo de processos para a bissimilaridade, e, como vimos, tem uma estrutura de Proc-álgebra.

1.4.7

Teorema. *A álgebra* STAA/$\sim$ *satisfaz as seguintes equações:*

1. $x + (y + z) = (x + y) + z$,
2. $x + \mathbf{0} = x$,
3. $x + y = y + x$,
4. $x + x = x$,
5. $x \parallel (y \parallel z) = (x \parallel y) \parallel z$,
6. $x \parallel \mathbf{0} = x$,
7. $x \parallel y = y \parallel x$,
8. $(\sum_{i=1}^{n} \alpha_i x_i) \parallel (\sum_{j=1}^{m} \beta_j y_j) =$

$$= \sum_{i=1}^{n} \alpha_i (x_i \parallel (\sum_{j=1}^{m} \beta_j y_j)) + \sum_{j=1}^{m} \beta_j ((\sum_{i=1}^{n} \alpha_i x_i) \parallel y_j)$$

$(m, n \geq 1)$.

[Nota: $\sum_{i=1}^{n} P_i \stackrel{\text{def}}{=} P_1 + \ldots + P_n$.]

Como se vê, as propriedades algébricas de STAA/$\sim$ são semelhantes às do modelo **T**, mas agora as equações que envolvem distributividade sobre a soma,

$$\begin{aligned} \alpha(x + y) &= \alpha x + \alpha y \,, \\ (x + y) \parallel z &= x \parallel z + y \parallel z \,, \end{aligned}$$

estão ausentes, e a equação

$$\alpha x \parallel \beta y = \alpha(x \parallel \beta y) + \beta(\alpha x \parallel y)$$

foi substituída por um conjunto numerável de equações (alínea 8 do Teorema). Por exemplo, em **T** podíamos provar

$$(\alpha + \beta) \parallel \gamma = \alpha\gamma + \beta\gamma + \gamma(\alpha + \beta)$$

aplicando as três equações acima, enquanto agora aplicamos directamente o caso $n = 2$, $m = 1$ da alínea 8 do Teorema.

1.4.8

Vimos como se pode descrever as propriedades algébricas de STAA/$\sim$ utilizando um conjunto numerável de axiomas. Vamos agora ver uma axiomatização alternativa com um número finito de axiomas (se Act for finito). Para tal introduzimos uma "operação auxiliar"representada pelo símbolo binário $\lfloor\!\lfloor$, chamado de *composição à esquerda*. A ideia é a de que $\mathsf{S} \lfloor\!\lfloor \mathsf{T}$ se comporta como $\mathsf{S} \parallel \mathsf{T}$, mas com a diferença de que T só pode executar alguma acção depois de S ter executado pelo menos uma.

Como axiomas utilizaremos os seguintes:

1. $x + (y + z) = (x + y) + z$,
2. $x + \mathbf{0} = x$,
3. $x + y = y + x$,
4. $x + x = x$,
5. $x \parallel y = x \lfloor\!\lfloor y + y \lfloor\!\lfloor x$,
6. $(x \lfloor\!\lfloor y) \lfloor\!\lfloor z = x \lfloor\!\lfloor (y \parallel z)$,
7. $\alpha x \lfloor\!\lfloor y = \alpha(x \parallel y)$,
8. $(x + y) \lfloor\!\lfloor z = x \lfloor\!\lfloor z + y \lfloor\!\lfloor z$,
9. $\mathbf{0} \lfloor\!\lfloor x = \mathbf{0}$,
10. $x \lfloor\!\lfloor \mathbf{0} = x$.

Exercício. Prove, utilizando estes axiomas, que

$$\begin{aligned} x \parallel (y \parallel z) &= (x \parallel y) \parallel z \ , \\ x \parallel \mathbf{0} &= x \ , \\ x \parallel y &= y \parallel x \ . \end{aligned}$$

Exercício. Defina uma interpretação apropriada para o símbolo $\between$ em STAA, de modo que $\sim$ seja uma congruência em STAA e os axiomas acima sejam satisfeitos em STAA/$\sim$.

1.4.9

Vamos agora examinar brevemente um modelo concreto de processos para bissimulação, para processos finitos. No que se segue ignoramos a operação de composição paralela.

Defina-se a família de conjuntos $\{B_n\}_{n\in\omega}$, onde $B_0 = \{\emptyset\}$ e, para cada $n \in \omega$, $B_{n+1} = 2^{Act \times B_n}$, e seja

$$\mathbf{B}_{\text{fin}} \stackrel{\text{def}}{=} \bigcup_{n\in\omega} B_n \ .$$

Os elementos de $\mathbf{B}_{\text{fin}}$ são conjuntos da forma $\{\langle\alpha, X\rangle, \langle\beta, Y\rangle, \ldots\}$, onde $\alpha, \beta, \ldots \in Act$ e $X, Y, \ldots \in \mathbf{B}_{\text{fin}}$.

Exercício. Verifique que $\mathbf{B}_{\text{fin}} = 2^{Act\times\mathbf{B}_{\text{fin}}}_{\text{fin}}$.

O conjunto $\mathbf{B}_{\text{fin}}$ é uma álgebra para a assinatura Proc^- cujos símbolos são os de Proc excepto $\parallel$: o processo nulo é $\emptyset$, a prefixação é dada por $\alpha X = \{\langle\alpha, X\rangle\}$ e a soma por $X + Y = X \cup Y$.

Exemplo. $(\alpha\beta + \gamma(\alpha + \beta))_{\mathbf{B}_{\text{fin}}}$ é o conjunto

$$\{\langle\alpha, \{\langle\beta, \emptyset\rangle\}\rangle, \langle\gamma, \{\langle\alpha, \emptyset\rangle, \langle\beta, \emptyset\rangle\}\rangle\} \ .$$

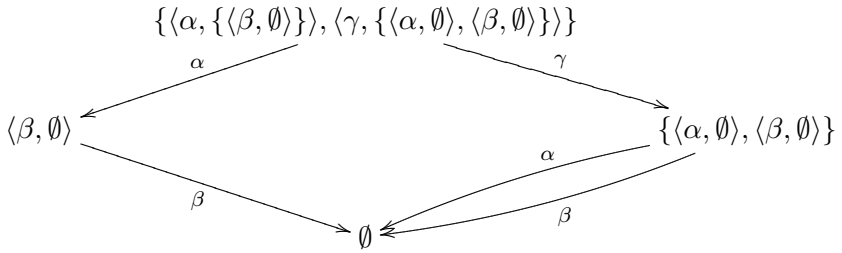
1.4.10

Podemos equipar o conjunto $\mathbf{B}_{\text{fin}}$ com uma relação de transição T, obtendo um sistema de transição $\langle\mathbf{B}_{\text{fin}}, T\rangle$, do seguinte modo:

$$X \stackrel{\alpha}{\to} Y \iff^{\text{def}} \langle\alpha, Y\rangle \in X \ .$$

Cada conjunto $X \in \mathbf{B}_{\text{fin}}$ dá portanto origem a um staa com estado inicial X, nomeadamente $\mathsf{staa}(\langle\mathbf{B}_{\text{fin}}, T, X\rangle)$, que designaremos também por X.

Exemplo. O conjunto $X = (\alpha\beta + \gamma(\alpha + \beta))_{\mathbf{B}_{\text{fin}}}$ obtido no Exemplo 1.4.9 tem o staa seguinte:



Teorema. *Para quaisquer $X, Y \in \mathbf{B}_{\text{fin}}$, tem-se $X \sim Y$ sse $X = Y$.*

Prova. Basta provar $X \sim Y \Rightarrow X = Y$, porque qualquer staa é bissimilar a si próprio. Defina-se, para cada $Z \in \mathbf{B}_{\text{fin}}$, $r(Z)$ como o menor n para o qual $Z \in B_n$. Vamos fazer a prova por indução em $r(X)$. Se $r(X) = 0$ então $X = \emptyset$, e $X \sim Y$ significa que Y não tem transições, ou, equivalentemente, não contém qualquer par ordenado da forma $\langle \alpha, Z \rangle$, o que significa que é vazio porque os elementos de Y são necessariamente pares ordenados desta forma. Suponha-se agora $r(X) \geq 1$. Se $X \xrightarrow{\alpha} X'$ então existe Y' tal que $Y \xrightarrow{\alpha} Y'$ e $X' \sim Y'$. Mas é fácil verificar que se $X \xrightarrow{\alpha} X'$ então $r(X') \leq r(X) - 1$, e portanto conclui-se $X' = Y'$, por hipótese de indução. Isto mostra, para qualquer $\alpha \in Act$ e $X' \in \mathbf{B}_{\text{fin}}$, que se $\langle \alpha, X' \rangle \in X$ então $\langle \alpha, X' \rangle \in Y$, ou seja, $X \subseteq Y$. Suponha-se agora $Y \xrightarrow{\alpha} Y'$. Então existe X' tal que $X \xrightarrow{\alpha} X'$ e $X' \sim Y'$. Novamente por hipótese de indução concluimos que $X' = Y'$ porque $r(X') \leq r(X) - 1$, e portanto $Y \subseteq X$. ∎

1.4.11

Seja $\mathsf{S} = \langle P, T, \imath \rangle$ um staa finito. Para cada estado $x \in P$ definimos um conjunto $\mathbf{B}(x)$ do modo seguinte:

$$\mathbf{B}(x) \stackrel{\text{def}}{=} \{\langle \alpha, Y \rangle \in Act \times \mathbf{B}_{\text{fin}} \mid \exists_{y \in P}((x \xrightarrow{\alpha} y) \wedge (\mathbf{B}(y) = Y))\} .$$

Esta é uma definição recursiva que se justifica porque o staa é por hipótese finito.

Exercício. Verifique esta última asserção. [Sugestão: a finitude de S implica que para cada estado x há um máximo $m(x) \in \omega$ para o comprimento dos traços t tais que $x \xrightarrow{t}$, e se $x \xrightarrow{\alpha} y$ então $m(y) < m(x)$.]

Agora a partir de S construimos o conjunto $\mathbf{B}(\mathsf{S}) \stackrel{\text{def}}{=} \mathbf{B}(\imath)$, e obtemos

Teorema. *1. A função* $\mathbf{B} : \mathrm{STAA}_{\mathrm{fin}} \to \mathbf{B}_{\mathrm{fin}}$ *é um homomorfismo, relativamente à assinatura* Proc^-.

2. Qualquer staa finito S *é bissimilar a* $\mathbf{B}(\mathsf{S})$.

[Nota: $\mathrm{STAA}_{\mathrm{fin}}$ é a subálgebra de STAA cujos elementos são os staas finitos.]

Corolário. *Sejam* $P, Q \in T_{\mathsf{Proc}^-}$. *Então* $P_{\mathrm{STAA}} \sim Q_{\mathrm{STAA}}$ *sse* $P_{\mathbf{B}_{\mathrm{fin}}} = P_{\mathbf{B}_{\mathrm{fin}}}$.

Estes resultados mostram que $\mathbf{B}_{\mathrm{fin}}$ é um bom domínio semântico para a bissimulação, no caso dos processos finitos.

1.4.12

Poderemos generalizar as construções anteriores para o caso de staas não finitos? A resposta em geral é negativa, pelo menos se desejarmos permanecer dentro da teoria de conjuntos "clássica", pela qual entendemos a teoria ZFC (v. por exemplo Johnstone [6]). Para ter uma breve ideia de um dos problemas envolvidos, suponha-se que é dado o staa

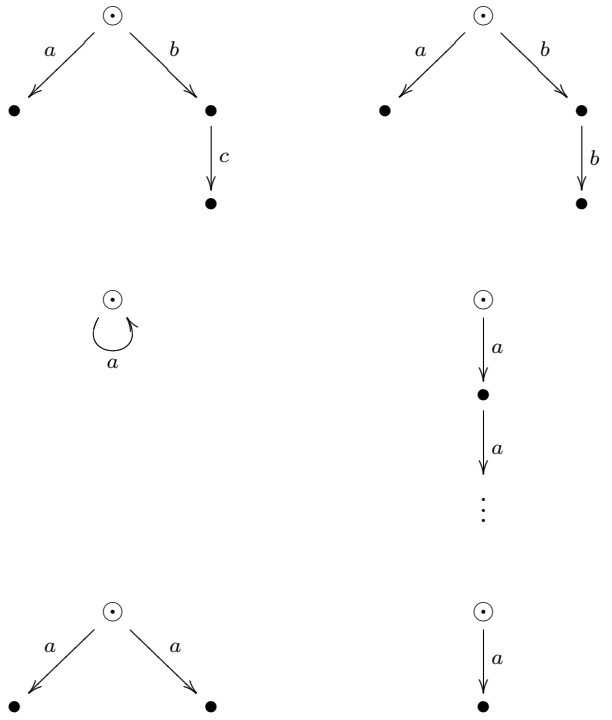
Se tentarmos daqui obter um conjunto de modo análogo ao que foi utilizado na definição recursiva da secção anterior, obteremos o "conjunto" $X = \{\langle \alpha, X \rangle\}$. Se admitirmos que existe um conjunto X nestas condições obteremos, segundo a representação habitual de pares ordenados em teoria de conjuntos,

$$X = \{\{\alpha\}, \{\alpha, X\}\} \ ,$$

e portanto X pertence a um conjunto ($\{\alpha, X\}$) que pertence a X. É simples ver que a existência deste conjunto viola o axioma da fundação. Um tratamento de conjuntos não-bem-fundados pode encontrar-se em Aczel [1], onde o axioma da fundação é substituído por um outro axioma, chamado de *anti-fundação*.

1.4.13 Exercícios

1. Quais dos seguintes pares de staas são bissimilares? Justifique.



2. Complete a prova do Lema 1.4.5.

3. [Completação da prova do Teorema 1.4.4.] Seja $\langle L, \leq \rangle$ um reticulado completo. Diz-se que uma função $j : L \to L$ é um *operador de fecho* se satisfaz as três condições seguintes, para quaisquer $x, y \in L$:

   - $x \leq j(x)$
   - $x \leq y \Rightarrow j(x) \leq j(y)$
   - $j(j(x)) = j(x)$

   Diz-se também que um elemento $x \in L$ é *fechado* (relativamente a j) se $x = j(x)$.

   (a) Justifique que um elemento $x \in L$ é fechado sse existe $y \in L$ tal que $x = j(y)$, e que portanto os elementos fechados de L são exactamente os elementos da imagem $j(L)$.

(b) Prove que qualquer operador de fecho j satisfaz a condição

$$x \leq j(y) \iff j(x) \leq j(y) \ .$$

(c) Prove que a função j da prova do Teorema 1.4.4 é um operador de fecho em $\langle 2^{P\times Q}, \subseteq\rangle$; isto é, que satisfaz, para quaisquer $S, S' \subseteq P \times Q$:

i. $S \subseteq j(S)$

ii. $S \subseteq S' \Rightarrow j(S) \subseteq j(S')$

iii. $j(j(S)) = j(S)$

(d) Prove que a função ϕ transforma elementos fechados em elementos fechados.

(e) Justifique que para qualquer $S \subseteq P \times Q$ se tem

$$S \subseteq \phi(j(S)) \iff j(S) \subseteq \phi(j(S)) \ .$$

1.5 Semântica operacional estrutural

No que se segue assumiremos que dispomos de um conjunto de geradores G e de um conjunto de relações $\rho \subseteq T_{\mathsf{Proc}}\langle G\rangle \times T_{\mathsf{Proc}}\langle G\rangle$, fixos.

Definimos agora uma relação de transição $\rightarrow$ sobre o conjunto de termos $T_{\mathsf{Proc}}\langle G\rangle$, como sendo o menor subconjunto de $T_{\mathsf{Proc}}\langle G\rangle \times Act \times T_{\mathsf{Proc}}\langle G\rangle$ fechado para as seguintes regras de inferência:

$$\textbf{Act} \quad \frac{}{\alpha P \xrightarrow{\alpha} P}$$

$$\textbf{Sum}_1 \quad \frac{P \xrightarrow{\alpha} P'}{P+Q \xrightarrow{\alpha} P'} \qquad \textbf{Sum}_2 \quad \frac{Q \xrightarrow{\alpha} Q'}{P+Q \xrightarrow{\alpha} Q'}$$

$$\textbf{Com}_1 \quad \frac{P \xrightarrow{\alpha} P'}{P \parallel Q \xrightarrow{\alpha} P' \parallel Q} \qquad \textbf{Com}_2 \quad \frac{Q \xrightarrow{\alpha} Q'}{P \parallel Q \xrightarrow{\alpha} P \parallel Q'}$$

$$\textbf{Con}_1 \quad \frac{\langle P,Q\rangle \in \rho \ , \ P \xrightarrow{\alpha} P'}{Q \xrightarrow{\alpha} P'} \qquad \textbf{Con}_2 \quad \frac{\langle P,Q\rangle \in \rho \ , \ Q \xrightarrow{\alpha} Q'}{P \xrightarrow{\alpha} Q'}$$

Cada regra $\frac{\Pi}{\varphi}$ deve ler-se "φ se Π"; Π é o conjunto de *premissas* e φ é a *conclusão* da regra. Por exemplo, a regra $\textbf{Sum}_1$ diz-nos que se para dois termos de processo P e P' se tem $P \xrightarrow{\alpha} P'$ então ter-se-á também $P + Q \xrightarrow{\alpha} P'$, para qualquer termo Q; a regra **Act**, que não tem premissas, diz-nos simplesmente que há uma transição $\alpha P \xrightarrow{\alpha} P$ para qualquer termo P.

O st assim definido,

$$SOS_{G,\rho} \stackrel{\text{def}}{=} \langle T_{\mathsf{Proc}}\langle G\rangle, \rightarrow\rangle \ ,$$

é designado por *semântica operacional estrutural* (SOS) da linguagem $T_{\mathsf{Proc}}\langle G\rangle$. [A expressão "SOS"vem do inglês "Structured Operational Semantics".]

Exercício. Mostre que se tem $abc + (d \parallel ef) \stackrel{e}{\rightarrow} d \parallel f$.

1.5.1

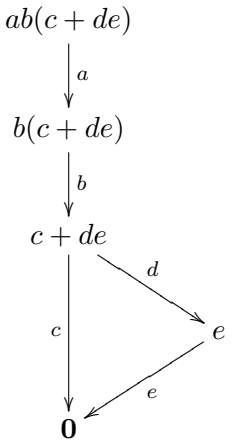
A SOS permite obter staas para especificações de processo dum modo diferente do que vimos anteriormente.

Definição. Seja P um termo de processo. Define-se o staa $SOS(P)$ como sendo $\mathsf{staa}(\mathsf{S})$, onde S é o sta que resulta de equipar $SOS_{G,\rho}$ com o estado inicial P.

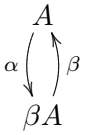
Notação. Normalmente escreveremos apenas P em vez de $SOS(P)$.

Exemplo.

1. $ab(c + de)$ tem o staa seguinte, com estado inicial $ab(c + de)$:



2. Suponha que o conjunto de geradores é $G = \{A\}$ e que existe uma única relação dada por $A = \alpha\beta A$. Então A tem o staa seguinte, com estado inicial A:



1.5.2

Teorema. *Seja* $P \in T_{\mathsf{Proc}}$. *Então* $P \sim P_{\mathrm{STAA}}$.

Prova. Por indução estrutural. ∎

Corolário. *Sejam* $P, Q \in T_{\mathsf{Proc}}$. *Então* $P \sim Q$ *sse* $P_{\mathrm{STAA}} \sim Q_{\mathrm{STAA}}$.

O Teorema mostra que a semântica algébrica vista anteriormente, em que cada termo de processo denota um elemento da álgebra STAA/$\sim$, é equivalente à SOS, no caso de termos de processo *sem* geradores. Um resultado análogo no caso geral com geradores é mais complicado de provar, envolvendo em particular a questão de quais especificações de processo têm solução em $STAA/\sim$, e não será abordado aqui.

1.5.3 Exercícios

1. Prove o Teorema 1.5.2.

2. Seja $G = \{C, C_0, C_1, C_2, \ldots\}$ e seja ρ o conjunto de relações seguintes:

$$\begin{aligned} C &= \mathtt{inc}.(C \parallel \mathtt{dec}) \\ C_0 &= \mathtt{inc}.C_1 \\ C_{n+1} &= \mathtt{inc}.C_{n+2} + \mathtt{dec}.C_n \ \ (n \geq 0) \end{aligned}$$

   (a) Desenhe (parte d)os staas $SOS(C)$ e $SOS(C_0)$.

   (b) Mostre que em $SOS_{G,\rho}$ se tem $C \sim C_0$. [Sugestão: é útil usar a noção de bissimulação a menos de $\sim$ da Secção 1.4.4.]

   Nota: Este exercício mostra que as especificações de contadores do exercício 1.3.3-10 são equivalentes se a equivalência de processos adoptada for a bissimilaridade (ou outra mais fraca).

Capítulo 2

Notas de 1998/99

Este capítulo contém adendas e correcções ao livro *Communication and Concurrency* de R. Milner [7].

2.1 Unicidade de pontos fixos (= soluções únicas de equações) módulo bissimilaridade forte

Definição. Seja $\mathcal{X}$ um conjunto de *variáveis*, disjunto de $\mathcal{K}$, e cujos elementos denotamos por X, Y, Z, X', X_1, etc. Uma *expressão sobre* $\mathcal{X}$ é um elemento de $\mathcal{P}(\mathcal{K} \cup \mathcal{X})$. Denotamos as expressões por E, F, G, E', F_1, etc.

Nota. Uma equação de definição continua a ser da forma $A \stackrel{\text{def}}{=} P$, onde $A \in \mathcal{K}$ e $P \in \mathcal{P}(\mathcal{K})$, e definimos uma relação de transição sobre o conjunto de expressões da maneira habitual. Em particular, resulta que das variáveis não partem quaisquer transições.

Notação. Escrevemos $E\{P/X\}$ para a expressão que resulta de substituir todas as ocorrências da variável X pelo agente P em E.

Definição. Uma variável X é *guardada* em E se ocorre sempre em subexpressões de E da forma $\alpha E'$. Diz-se que uma expressão é *guardada* se todas as suas variáveis são guardadas.

O objectivo é provar o seguinte teorema:

Teorema. *Seja I um conjunto e sejam dadas as seguintes famílias:*

- *de agentes $\tilde{P} = \{P_i\}_{i \in I}$*

- *de agentes* $\tilde{Q} = \{Q_i\}_{i \in I}$
- *de variáveis* $\tilde{X} = \{X_i\}_{i \in I}$
- *de expressões guardadas* $\tilde{E} = \{E_i\}_{i \in I}$*, sobre as variáveis de* $\tilde{X}$

Se para cada $i \in I$ *se tiver*

$$\begin{aligned} P_i &\sim E_i(\tilde{P}) \\ Q_i &\sim E_i(\tilde{Q}) \end{aligned}$$

então $\tilde{P} \sim \tilde{Q}$ (i.e., $P_i \sim Q_i$ *para qualquer* $i \in I$*).*

[Nota: escrevemos $E(\tilde{P})$ para representar a substituição em E de X_i por P_i para todos os valores $i \in I$.]

Para provar este teorema recorremos primeiro ao seguinte lema, que nos diz que numa expressão guardada a primeira transição não depende dos agentes com os quais instanciamos as variáveis:

Lema. *Seja* E *uma expressão guardada sobre as variáveis duma família contável* $\tilde{X} = \{X_i\}_{i \in I}$ *de variáveis, e seja* $\tilde{P} = \{P_i\}_{i \in I}$ *uma família de agentes. Então, se* $E(\tilde{P}) \xrightarrow{\alpha} P'$ *tem-se* $P' \equiv E'(\tilde{P})$ *para alguma expressão (não necessariamente guardada) sobre* $\tilde{X}$ *e, para qualquer família* $\tilde{Q} = \{Q_i\}_{i \in I}$ *de agentes, há uma transição* $E(\tilde{Q}) \xrightarrow{\alpha} E'(\tilde{Q})$*.*

Prova. Seja E uma expressão nas condições do lema, tal que $E(\tilde{P}) \xrightarrow{\alpha} P'$. A prova será por indução na estrutura de E.

1. Caso $E \equiv Y$, Y uma variável. Então $Y \notin \tilde{X}$ porque por hipótese E é guardada. Mas então $E(\tilde{P}) \equiv Y$ não tem quaisquer transições, um absurdo, o que mostra que E não pode ser uma variável.

2. Caso $E \equiv \mathbf{0}$. Este caso também é impossível porque $\mathbf{0}$ não tem transições.

3. Caso $E \equiv A$, A uma constante, onde $A \stackrel{\text{def}}{=} R$ (R um agente). Então $A \xrightarrow{\alpha} P'$ se e só se $R \xrightarrow{\alpha} P'$; além disso um tal P' não tem variáveis e portanto faz-se $E' \equiv P'$.

4. Caso $E \equiv \beta.F$, F uma expressão, $\beta \in Act$. Agora tem-se $E(\tilde{P}) \xrightarrow{\alpha} P'$ se e só se $\alpha = \beta$ e $P' \equiv F(\tilde{P})$. Então o resultado obtém-se fazendo $E' \equiv F$.

5. Caso $E \equiv E_1 + E_2$. Então $E_i(\tilde{P}) \xrightarrow{\alpha} P'$ para algum $i = 1, 2$. Mas E_i é subexpressão de E, e por hipótese de indução concluímos que $P' \equiv E'(\tilde{P})$ para alguma expressão E', e que $E_i(\tilde{Q}) \xrightarrow{\alpha} E'(\tilde{Q})$. Logo, $E(\tilde{Q}) \xrightarrow{\alpha} E'(\tilde{Q})$.

6. Caso $E \equiv E_1 \,|\, E_2$. Agora há três casos possíveis:

   (a) $P' \equiv P'' \,|\, E_2(\tilde{P})$, com $E_1(\tilde{P}) \xrightarrow{\alpha} P''$. Como E_1 é subexpressão de E conclui-se, pela hipótese de indução, que P'' é da forma $E_1'(\tilde{P})$ e que $E_1(\tilde{Q}) \xrightarrow{\alpha} E_1(\tilde{Q})$. Mas então, fazendo $E' \equiv E_1' \,|\, E_2$ obtém-se $P' \equiv E'(\tilde{P})$ e $E(\tilde{Q}) \xrightarrow{\alpha} E'(\tilde{Q})$.

   (b) $P' \equiv E_1(\tilde{P}) \,|\, P''$, com $E_2(\tilde{P}) \xrightarrow{\alpha} P''$. Este caso é análogo ao anterior.

   (c) $P' \equiv P_1' \,|\, P_2'$, onde $E_1(\tilde{P}) \xrightarrow{\ell} P_1'$ e $E_2(\tilde{P}) \xrightarrow{\bar{\ell}} P_2'$, para alguma etiqueta ℓ. Agora, por hipótese de indução, existem expressões E_1' e E_2' tais que $P_i' \equiv E_i'(\tilde{P})$ para $i = 1, 2$, e tais que $E_1(\tilde{P}) \xrightarrow{\ell} E_1'(\tilde{P})$ e $E_2(\tilde{P}) \xrightarrow{\bar{\ell}} E_2'(\tilde{P})$. Então basta fazer $E' \equiv E_1' \,|\, E_2'$.

7. Caso $E \equiv F\backslash L$. Agora tem-se $F(\tilde{P}) \xrightarrow{\alpha} P''$, com $P' \equiv P''\backslash L$. Por hipótese de indução, $P'' \equiv F(\tilde{P})$ e $F(\tilde{Q}) \xrightarrow{\alpha} F(\tilde{Q})$, donde fazemos $E' \equiv F\backslash L$.

8. Caso $E \equiv F[f]$. Agora tem-se $F(\tilde{P}) \xrightarrow{\beta} P''$, com $P' \equiv P''[f]$ e $f(\beta) = \alpha$. Por hipótese de indução, $P'' \equiv F(\tilde{P})$ e $F(\tilde{Q}) \xrightarrow{\beta} F(\tilde{Q})$, donde fazemos $E' \equiv F[f]$. ∎

Prova do teorema:

Prova. Queremos mostrar que para cada $i \in I$ se tem $P_i \sim Q_i$, e para tal vamos mostrar que a relação

$$R = \{\langle E(\tilde{P}), E(\tilde{Q})\rangle \mid \mathit{Vars}(E) \subseteq \tilde{X}\}$$

é uma bissimulação a menos de bissimilaridade. (Note-se que a expressão E acima é qualquer, desde que as suas variáveis sejam todas escolhidas de $\tilde{X}$, e portanto inclui o caso $E \equiv X_i$, caso em que se obtém $P_i R Q_i$, o que nos permitirá concluir $P_i \sim Q_i$.) Por simetria basta provar

$$E(\tilde{P}) \xrightarrow{\alpha} P' \Rightarrow \exists_{Q'}(E(\tilde{Q}) \xrightarrow{\alpha} Q' \ \wedge \ P' \sim R \sim Q') \ .$$

Assuma-se então $E(\tilde{P}) \stackrel{\alpha}{\rightarrow} P'$. Como por hipótese $\tilde{P} \sim \tilde{E}(\tilde{P})$ e a bissimilaridade forte é uma congruência para todas as operações utilizadas para construir E (que não contém variáveis livres depois de feita a substituição), resulta $E(\tilde{P}) \sim E(\tilde{E}(\tilde{P}))$. Portanto $E(\tilde{E}(\tilde{P})) \stackrel{\alpha}{\rightarrow} P''$ para algum P'' tal que $P' \sim P''$. Agora note-se que $E(\tilde{E}(\tilde{P})) \equiv E(\tilde{E})(\tilde{P})$ e que a expressão $E(\tilde{E})$ é guardada. Logo, pelo lema concluimos que $P'' \equiv E'(\tilde{P})$ para alguma expressão E' com variáveis de $\tilde{X}$, e que $E(\tilde{E})(\tilde{Q}) \stackrel{\alpha}{\rightarrow} E'(\tilde{Q})$. Agora, como por hipótese $\tilde{E}(\tilde{Q}) \sim \tilde{Q}$, e novamente por $\sim$ ser congruência, deve existir Q' bissimilar a $E'(\tilde{Q})$ tal que $E(\tilde{Q}) \stackrel{\alpha}{\rightarrow} Q'$. Portanto tem-se

$$P' \ \sim \ E'(\tilde{P}) \ R \ E'(\tilde{Q}) \ \sim \ Q' \ ,$$

o que termina a prova. ∎

2.2 Unicidade de pontos fixos (= soluções únicas de equações) módulo congruência observacional

No que se segue, $\tilde{V}$ denota uma família $\{V_i\}_{i \in I}$, onde I é um conjunto fixo. Expressões como $\tilde{P}/\tilde{X}$ denotam a substituição de X_i por P_i para qualquer $i \in I$, e expressões como $\tilde{P} \simeq \tilde{Q}$ denotam a afirmação de que $P_i \simeq Q_i$ para qualquer $i \in I$; uma expressão como $X \in \tilde{X}$ significa $X = X_i$ para algum $i \in I$.

Lema. *Sejam $P, Q \in \mathcal{P}(\mathcal{K})$. Então tem-se $P \simeq Q$ se e só se ambas as condições seguintes se verificam, para quaisquer agentes P' e Q':*

$$\begin{array}{rcl} P \stackrel{\alpha}{\Rightarrow} P' & \Rightarrow & \exists_{Q'}(Q \stackrel{\alpha}{\Rightarrow} Q' \ \wedge \ P' \approx Q') \\ Q \stackrel{\alpha}{\Rightarrow} Q' & \Rightarrow & \exists_{P'}(P \stackrel{\alpha}{\Rightarrow} P' \ \wedge \ P' \approx Q') \end{array}$$

Prova. Exercício. ∎

Lema. *Sejam $P, Q \in \mathcal{P}(\mathcal{K})$. Se $P \simeq Q$ então, para quaisquer agentes P' e Q',*

$$\begin{array}{rcl} P \stackrel{\varepsilon}{\Rightarrow} P' & \Rightarrow & \exists_{Q'}(Q \stackrel{\varepsilon}{\Rightarrow} Q' \ \wedge \ P' \approx Q') \\ Q \stackrel{\varepsilon}{\Rightarrow} Q' & \Rightarrow & \exists_{P'}(P \stackrel{\varepsilon}{\Rightarrow} P' \ \wedge \ P' \approx Q') \end{array}$$

Prova. Exercício. ∎

Lema. (Milner, Lemma 7.12) *Seja E uma expressão fortemente guardada e sequencial com variáveis tiradas apenas da sequência $\tilde{X}$, e seja $E(\tilde{P}) \xrightarrow{\alpha} P'$. Então existe uma expressão F com variáveis só de $\tilde{X}$ tal que $P' \equiv F(\tilde{P})$ e tal que para qualquer $\tilde{Q}$ se verifica $E(\tilde{Q}) \stackrel{\alpha}{\Rightarrow} F(\tilde{Q})$. Além disto, F é necessariamente sequencial e, se $\alpha = \tau$, é guardada.*

Lema. *Seja E uma expressão fortemente guardada e sequencial com variáveis tiradas apenas da sequência $\tilde{X}$, e seja $E(\tilde{P}) \Rightarrow \xrightarrow{\alpha} P'$. Então existe uma expressão F com variáveis só de $\tilde{X}$ tal que $P' \equiv F(\tilde{P})$ e tal que para qualquer $\tilde{Q}$ se verifica $E(\tilde{Q}) \Rightarrow \xrightarrow{\alpha} F(\tilde{Q})$. Além disto, F é necessariamente sequencial.*

Prova. Assuma-se $E(\tilde{P}) \xrightarrow{\tau^n} \xrightarrow{\alpha} P'$ ($n \geq 0$). A prova é por indução em n. A base é o caso $n = 0$, que resulta imediatamente do lema anterior. Se $n > 0$ então temos, também pelo lema anterior, $E(\tilde{P}) \xrightarrow{\tau} F'(\tilde{P}) \xrightarrow{\tau^{n-1}} \xrightarrow{\alpha} P'$ com F' fortemente guardada e sequencial, de tal maneira que $E(\tilde{Q}) \xrightarrow{\tau} F'(\tilde{Q})$. Por hipótese de indução resulta que $P' \equiv F(\tilde{P})$ para alguma expressão sequencial F, e que $F'(\tilde{Q}) \Rightarrow \xrightarrow{\alpha} F(\tilde{Q})$, pelo que $E(\tilde{Q}) \Rightarrow \xrightarrow{\alpha} F(\tilde{Q})$. ∎

Teorema. *Seja $\tilde{E} = \{E_i\}_{i \in I}$ uma família de expressões fortemente guardadas e sequenciais, com variáveis livres tiradas de $\tilde{X} = \{X_i\}_{i \in I}$, e sejam $\tilde{P} = \{P_i\}_{i \in I}$ e $\tilde{Q} = \{Q_i\}_{i \in I}$ famílias de agentes tais que*

$$\begin{aligned} \tilde{P} &\simeq \tilde{E}\{\tilde{P}/\tilde{X}\} \\ \tilde{Q} &\simeq \tilde{E}\{\tilde{Q}/\tilde{X}\} \end{aligned}$$

Então $\tilde{P} \simeq \tilde{Q}$.

Prova. Queremos mostrar que para cada $i \in I$ se tem $P_i \simeq Q_i$, e para tal vamos mostrar que a relação

$$\rho = \{\langle E(\tilde{P}), E(\tilde{Q})\rangle \mid \mathit{Vars}(E) \subseteq \tilde{X} \text{ e } E \text{ é sequencial}\}$$

satisfaz a condição

$$E(\tilde{P}) \stackrel{\alpha}{\Rightarrow} P' \Rightarrow \exists_{Q'}(E(\tilde{Q}) \stackrel{\alpha}{\Rightarrow} Q' \ \wedge \ P' \approx \rho \approx Q') \ .$$

Por simetria resultará que ρ é uma bissimulação fraca a menos de bissimilaridade fraca, donde $\approx\rho\approx$ é uma bissimulação fraca, e portanto ter-se-á $E(\tilde{P}) \simeq E(\tilde{Q})$ (note-se que E é uma expressão sequencial qualquer, desde que as suas variáveis sejam todas escolhidas de $\tilde{X}$, e portanto inclui o caso $E \equiv X_i$, caso em que se obtém $P_i \ \rho \ Q_i$, o que nos permitirá concluir

$P_i \simeq Q_i$.) Assuma-se então $E(\tilde{P}) \stackrel{\alpha}{\Rightarrow} P'$. Como por hipótese $\tilde{P} \simeq \tilde{E}(\tilde{P})$ e a congruência observacional é uma congruência para todas as operações utilizadas para construir E (que não contém variáveis livres depois de feita a substituição), resulta $E(\tilde{P}) \simeq E(\tilde{E}(\tilde{P}))$. Portanto $E(\tilde{E}(\tilde{P})) \stackrel{\alpha}{\Rightarrow} P''$ para algum P'' tal que $P' \approx P''$. Assuma-se $E(\tilde{E}(\tilde{(P)}) \Rightarrow \stackrel{\alpha}{\rightarrow} P''' \Rightarrow P''$, e note-se que $E(\tilde{E}(\tilde{P})) \equiv E(\tilde{E})(\tilde{P})$ e que a expressão $E(\tilde{E})$ é fortemente guardada, porque todas as expressões de $\tilde{E}$ o são, e sequencial porque todas estas são e E também é. Logo, pelo lema anterior concluimos que P''' deve ser da forma $E'(\tilde{P})$ com E' sequencial e que $E(\tilde{E}(\tilde{Q})) \Rightarrow \stackrel{\alpha}{\rightarrow} E'(\tilde{Q})$. Agora há dois casos: (1) se $P'' \equiv P'''$, de $E(\tilde{E}(\tilde{Q})) \simeq E(\tilde{Q})$ obtemos que existe Q' tal que $E(\tilde{Q}) \stackrel{\alpha}{\Rightarrow} Q'$ e $P'' \approx Q'$, onde se tem $P' \approx P''$ ρ $E'(\tilde{Q}) \approx Q'$, e portanto $P' \approx\rho\approx Q'$ como pretendido. Caso (2): Se $P''' \stackrel{\tau}{\Rightarrow} P''$ temos, porque $E'(\tilde{P}) \simeq E'(\tilde{E}(\tilde{P}))$, que existe E'' tal que $P'' \approx E''(\tilde{P})$ e $E'(\tilde{E}(\tilde{P})) \stackrel{\tau}{\Rightarrow} E''(\tilde{P})$ e, novamente pelo lema anterior, tal que $E'(\tilde{E}(\tilde{Q})) \stackrel{\tau}{\Rightarrow} E''(\tilde{Q})$, pois $E'(\tilde{E})$ é sequencial e fortemente guardada. Mas então tem-se $E(\tilde{E}(\tilde{Q})) \stackrel{\alpha}{\Rightarrow} E''(\tilde{Q})$, e portanto, como $E(\tilde{E}(\tilde{Q})) \simeq E(\tilde{Q})$, resulta que existe Q' tal que $E(\tilde{Q}) \stackrel{\alpha}{\Rightarrow} Q'$ e $E''(\tilde{Q}) \approx Q'$; resumindo, temos $P' \approx P'' \approx E''(\tilde{P})$ ρ $E''(\tilde{Q}) \approx Q'$, donde $P' \approx\rho\approx Q'$, o que conclui a demonstração. ∎

2.3 Sistemas com número finito de estados

Proposição. *Se $P \in \mathcal{P}(\emptyset)$ então o sta de P tem um número finito de estados.*

Prova. Por indução estrutural:

1. $P \equiv \mathbf{0}$—Trivial.
2. $P \equiv \alpha.Q$—Os estados são P e os estados do sta de Q, que são em número finito por hipótese de indução.
3. $P \equiv P_1 + P_2$—Os estados são $P_1 + P_2$ e todos os estados acessíveis a partir de P_1 e P_2, que por hipótese de indução são em número finito.
4. $P \equiv P_1 \mid P_2$—Por hipótese de indução os números de estados de P_1 e de P_2 são finitos. Sejam eles n_1 e n_2. É fácil ver que o número de estados de $P_1 \mid P_2$ é necessariamente menor ou igual a $n_1 \times n_2$.
5. $P \equiv Q\backslash L$—É fácil de ver que o número de estados de P é inferior ou igual ao número de estados de Q, que por hipótese de indução é finito.
6. $P \equiv Q[f]$—Análogo ao caso anterior. ∎

Teorema. *Sejam* $\tilde{A} \stackrel{def}{=} \tilde{E}\{\tilde{A}/\tilde{X}\}$ *n equações de definição* ($n \in \omega$)*, onde as expressões* $\tilde{E}$ *são sequenciais com variáveis em* $\tilde{X}$*. Então o número de estados de* A_1 *é finito.*

[Antes de provar este teorema convém dar um exemplo para motivar o porquê do conjunto Σ da prova abaixo; por exemplo $A \stackrel{\text{def}}{=} \alpha\beta A + \alpha(A + B) + \gamma \,|\, \alpha$ e $B \stackrel{\text{def}}{=} \beta\beta A + A + B$.]

Prova. Seja

$$\Sigma = \{A_1, \ldots, A_n\} \cup \{\text{subagentes de algum } E_i\{\tilde{A}/\tilde{X}\}\ \} \cup$$
$$\cup \{\text{estados dos stas dos subagentes em que não ocorre nenhum } A_i\}\ .$$

Note-se que o conjunto Σ é finito, pois o número de subagentes envolvidos é necessariamente finito, e para cada agente em que não surge nenhuma constante o respectivo sta tem necessariamente um número finito de estados, pela proposição anterior. Vamos mostrar que Σ é fechado para transições, concluindo portanto a demonstração. Seja $P \in \Sigma$, tal que $P \stackrel{\alpha}{\to} P'$. Mostraremos que necessariamente $P' \in \Sigma$, por indução na maior profundidade das derivações de $P \stackrel{\alpha}{\to} P'$.

1. Primeiro assumimos que em P ocorre alguma constante A_i, sendo P portanto A_i ou subagente de algum agente $E_i\{\tilde{A}/\tilde{X}\}$. Note-se também que P não pode ser da forma $P_1 \,|\, P_2$, $Q\backslash L$ ou $Q[f]$, porque por hipótese as expressões são sequenciais, o que nos conduz aos casos seguintes.
   (a) Se $P \equiv A_i$ então $E_i\{\tilde{A}/\tilde{X}\} \stackrel{\alpha}{\to} P'$. Como $E_i\{\tilde{A}/\tilde{X}\} \in \Sigma$, por hipótese de indução concluímos que $P' \in \Sigma$.
   (b) Se $P \equiv \beta.P''$ então $\alpha = \beta$ e $P'' \equiv P'$, pelo que P'' é subagente de P, e portanto $P'' \in \Sigma$.
   (c) Se $P \equiv P_1 + P_2$ então $P_i \stackrel{\alpha}{\to} P'$ para algum $i = 1, 2$, e por hipótese de indução obtém-se $P' \in \Sigma$.
2. Agora assumimos que em P não ocorre nenhuma constante A_i. Portanto P é um estado dum subagente de algum $E_i\{\tilde{A}/\tilde{X}\}$ em que não ocorrem constantes, e todos os estados acessíveis a partir dele estão também em Σ, por construção de Σ. Logo, $P' \in \Sigma$. ∎

2.4 Lógica de Hennessy e Milner (HML)

A lógica de Hennessy e Milner (HML) é um modo de generalizar a equivalência de traços a fim de obter equivalências mais fortes. O objectivo é mesmo mostrar que a bissimilaridade forte pode ser definida como uma equivalência de "traços generalizados"em que

$$p \sim q \iff \text{paraqualquer"tracogeneralizado"}\varphi,\ p \xrightarrow{\varphi} \Leftrightarrow q \xrightarrow{\varphi}.$$

O objectivo deste capítulo é analisar em que medida isto é possível.

Primeiro recorde-se que a relação "um estado x dum st tem o traço t", que escrevemos $x \xrightarrow{t}$, pode ser definida indutivamente por:

- $x \xrightarrow{\varepsilon}$
- $x \xrightarrow{\alpha t}$ se e só se existe y tal que $x \xrightarrow{\alpha} y$ e $y \xrightarrow{t}$

Os "traços generalizados", a que chamamos *fórmulas* (de HML), generalizam simultaneamente a noção de traço e a de fórmula proposicional:

Definição. As *fórmulas de HML* são definidas indutivamente como se segue.

- $\top$ é uma fórmula.
- Se φ e ψ são fórmulas então $\neg\varphi$ e $\varphi \wedge \psi$ são fórmulas.
- Se φ é uma fórmula e α é uma acção então $\langle\alpha\rangle\varphi$ é uma fórmula.

O conjunto das fórmulas é denotado por $\mathcal{L}_{\text{HML}}$.

Intuitivamente, $\top$ representa o "traço generalizado"vazio ε e $\langle\alpha\rangle\varphi$ representa o "traço generalizado"$\alpha\varphi$. A relação "$x \xrightarrow{\varphi}$"é agora escrita $x \models \varphi$ e é definida como se segue:

Definição. Seja $\langle P, \rightarrow\rangle$ um st e $x \in P$. Definimos uma relação $\vDash \subseteq P \times \mathcal{L}_{\text{HML}}$ indutivamente:

- $x \models \top$.
- $x \models \neg\varphi$ se e só se $x \not\models \varphi$.
- $x \models \varphi \wedge \psi$ se e só se $x \models \varphi$ e $x \models \psi$.
- $x \models \langle\alpha\rangle\varphi$ se e só se existe um estado $y \in P$ tal que $x \xrightarrow{\alpha} y$ e $y \models \varphi$.

Quando $x \models \varphi$ dizemos que x *satisfaz* φ, e a relação $\vDash$ é designada por *relação de sastisfação.*

Intuitivamente, a satisfação $x \models \varphi$ é a versão generalizada de x ter o "traço"φ, mas também é uma generalização da satisfação habitual da lógica proposicional.

A equivalência de "traços generalizados"é designada por *equivalência lógica* e define-se como se segue.

Definição. Dois estados x e y dum st são *logicamente equivalentes* se para qualquer fórmula φ se tem $x \models \varphi \Leftrightarrow y \models \varphi$, e escrevemos $x \sim_{\text{HML}} y$.

Para além das fórmulas definidas acima, também se utilizam outras como abreviaturas das primeiras:

Definição.

$$\begin{aligned} \bot &= \neg\top \\ \varphi \vee \psi &= \neg(\neg\varphi \wedge \neg\psi) \\ [\alpha]\varphi &= \neg\langle\alpha\rangle\neg\varphi \end{aligned}$$

Exercício. Mostre que num st $\langle P, \rightarrow\rangle$ se tem, para qualquer $x \in P$

1. $x \not\models \bot$.
2. $x \models \varphi \vee \psi$ se e só se $x \models \varphi$ ou $x \models \psi$.
3. $x \models [\alpha]\varphi$ se e só se para qualquer $y \in P$ tal que $x \xrightarrow{\alpha} y$ se tem $y \models \varphi$.

Vamos agora iniciar a comparação entre a equivalência lógica e a bissimilaridade forte.

Proposição. *Sejam x e y estados dum st. Se $x \sim y$ então $x \sim_{\text{HML}} y$.*

Prova. Feita nas aulas (por indução na estrutura das fórmulas). ∎

Portanto a bissimilaridade forte continua a ser pelo menos tão forte quanto a nova "equivalência de traços". Verificar-se-á a igualdade? Em geral não, como veremos adiante, mas nalguns casos (bastante gerais) sim. Comecemos por examinar esses casos, para os quais necessitamos de algumas definições adicionais.

Definição. Seja $\langle P, \rightarrow \rangle$ um st, $X \subseteq P$ e $y \in P$. Dizemos que y é *aderente* a X se para qualquer fórmula φ tal que $x \models \varphi$ existe um estado $x \in X$ tal que $x \models \varphi$. Ao conjunto de todos os pontos aderentes a X chama-se a *aderência* de X, denotada por $\overline{X}$. Se $X = \overline{X}$ então o conjunto X diz-se *fechado*.

Note-se a intuição topológica que está patente nesta definição, se pensarmos nas fórmulas como conjuntos abertos e na satisfação $x \models \varphi$ como uma forma de dizer que x "pertence"ao "conjunto aberto"φ: um ponto y é aderente a um conjunto X quando qualquer das suas "vizinhanças"(i.e., fórmulas que ele satisfaz) "intersecta"X. Na verdade, esta ideia é mais do que intuitiva, pois de facto poder-se-ia ver que esta noção de aderência define um *espaço topológico* no sentido usual.

Vamos agora examinar algumas propriedades da aderência.

Proposição. *Seja $\langle P, \rightarrow \rangle$ um st e $X, Y \subseteq P$.*

1. $\overline{\emptyset} = \emptyset$.
2. *Se $X \subseteq Y$ então $\overline{X} \subseteq \overline{Y}$ (monotonia).*
3. $X \subseteq \overline{X}$.
4. $\overline{\overline{X}} = \overline{X}$ *(idempotência).*
5. $\overline{X \cup Y} = \overline{X} \cup \overline{Y}$ *(aditividade finita).*
6. *Se X é finito então $\overline{X} = \bigcup_{x \in X} \overline{\{x\}}$.*

Prova. Vamos provar apenas a aditividade finita; as outras propriedades deixam-se como exercício. Da monotonia resulta, uma vez que $X \subseteq X \cup Y$ e $Y \subseteq X \cup Y$, que $\overline{X} \cup \overline{Y} \subseteq \overline{X \cup Y}$, portanto vamos apenas provar $\overline{X \cup Y} \subseteq \overline{X} \cup \overline{Y}$. Na verdade vamos provar o recíproco, i.e., vamos mostrar que para qualquer estado $x \in P$ se $x \notin \overline{X} \cup \overline{Y}$ então $x \notin \overline{X \cup Y}$. Seja então $x \notin \overline{X}$ e $x \notin \overline{Y}$. Por definição de aderência existem fórmulas φ e ψ tais que $x \models \varphi$ e $x \models \psi$ e tais que para qualquer $y \in X$ se tem $y \not\models \varphi$ e para qualquer $y \in Y$ se tem $y \not\models \psi$. Portanto tem-se $x \models \varphi \wedge \psi$ e, para qualquer $y \in X \cup Y$, $y \not\models \varphi \wedge \psi$, e portanto $x \notin \overline{X \cup Y}$. ∎

Definição. Seja $\langle P, \rightarrow \rangle$ um st e $X \subseteq P$. O conjunto X diz-se *pseudo-fechado* se $\overline{X} = \bigcup_{x \in X} \overline{\{x\}}$. O st é *de imagens pseudo-fechadas* se para qualquer $x \in P$ e qualquer acção α o conjunto $x \cdot \alpha$ $(= \{y \in P \mid x \xrightarrow{\alpha} y\})$ é pseudo-fechado.

Da última alínea da proposição anterior resulta imediatamente que qualquer conjunto finito é pseudo-fechado e portanto que qualquer st de imagens finitas é também de imagens pseudo-fechadas.

Vamos agora mostrar que a equivalência lógica coincide com a bissimilaridade forte para sts de imagens pseudo-fechadas (e portanto também para sts de imagens finitas).

Teorema. *Seja $\langle P, \rightarrow \rangle$ um st de imagens pseudo-fechadas, $x, y \in P$. Se $x \sim_{\text{HML}} y$ então $x \sim y$.*

Prova. Vamos mostrar que $\sim_{\text{HML}}$ é uma bissimulação forte em $\langle P, \rightarrow \rangle$ (logo, $\sim_{\text{HML}} \subseteq \sim$). Uma vez que $\sim_{\text{HML}}$ é uma relação de equivalência (verifique), então em particular é simétrica e por isso basta provar que é uma simulação, ou seja, que se $x \sim_{\text{HML}} y$ e $x \xrightarrow{\alpha} x'$ então existe y' tal que $y \xrightarrow{\alpha} y'$ e $x' \sim_{\text{HML}} y'$. Seja então $x \sim_{\text{HML}} y$ e $x \xrightarrow{\alpha} x'$. Seja φ uma fórmula arbitrária tal que $x' \models \varphi$. Então $x \models \langle\alpha\rangle\varphi$ e, como $x \sim_{\text{HML}} y$, também $y \models \langle\alpha\rangle\varphi$. Logo, existe y'' tal que $y \xrightarrow{\alpha} y''$ e $y'' \models \varphi$. Como φ é uma fórmula arbitrária, concluímos que $x' \in \overline{y \cdot \alpha}$. Mas como por hipótese $y \cdot \alpha$ é pseudo-fechado concluímos que existe $y' \in y \cdot \alpha$ (i.e., y' tal que $y \xrightarrow{\alpha} y'$) tal que $x' \in \overline{\{y'\}}$. Esta última condição significa que se $x' \models \psi$ então $y' \models \psi$, para qualquer fórmula ψ. Daqui resulta também que se $y' \models \psi$ (i.e., $y' \not\models \neg\psi$) então $x' \models \psi$ (i.e., $x' \not\models \neg\psi$), e portanto $x' \sim_{\text{HML}} y'$. Uma vez que $y \xrightarrow{\alpha} y'$, conclui-se que $\sim_{\text{HML}}$ é uma simulação, como pretendido. ∎

Note-se que qualquer sistema com um número finito de estados é de imagens finitas e portanto a bissimilaridade coincide com a equivalência lógica, o que inclui um grande número de sistemas de interesse na prática. Também o st do CCS é de imagens finitas se nos restringirmos a somas finitas (verifique), e portanto a bissimilaridade coincide com a equivalência lógica para qualquer par de agentes CCS.

Vamos finalmente mostrar que a bissimilaridade não coincide com a equivalência lógica todos os sistemas. Para tal serão úteis algumas definições auxiliares.

Definição. Seja $\langle P, \rightarrow \rangle$ um st. Definimos uma família de relações binárias $\{\sim_n\}_{n \in \omega}$ sobre P indutivamente como se segue.

- $\sim_0 = P \times P$.
- $x \sim_{n+1} y \overset{\text{def}}{\iff} \begin{cases} x \xrightarrow{\alpha} x' \Rightarrow \exists_{y'}(y \xrightarrow{\alpha} y' \;\&\; x' \sim_n y') \\ y \xrightarrow{\alpha} y' \Rightarrow \exists_{x'}(x \xrightarrow{\alpha} x' \;\&\; x' \sim_n y') \end{cases}.$

Definimos também a relação $\sim_\omega = \bigcap_{n\in\omega} \sim_n$.

Por outras palavras, $\sim_{n+1} = \Phi(\sim_n)$, onde $\Phi : 2^{P\times P} \to 2^{P\times P}$ é o habitual operador monótono que se utiliza para definir bissimulação (i.e., tal que R é uma bissimulação se e só se $R \subseteq \Phi(R)$).

Proposição. *As relações $\sim_\lambda$ satisfazem as seguintes propriedades.*

1. $\sim_{n+1} \subseteq \sim_n$ *para qualquer* $n \in \omega$.
2. $\sim \subseteq \sim_\omega$.

Prova. Exercício (sugestão: utilize a monotonia do operador Φ e indução em n). ∎

Definição. Define-se a *profundidade modal* das fórmulas de HML, $\mathrm{pm} : \mathcal{L}_{\mathrm{HML}} \to \omega$, indutivamente da seguinte forma.

- $\mathrm{pm}(\top) = 0$.
- $\mathrm{pm}(\neg\varphi) = \mathrm{pm}(\varphi)$.
- $\mathrm{pm}(\varphi \wedge \psi) = \max\{\mathrm{pm}(\varphi), \mathrm{pm}(\psi)\}$.
- $\mathrm{pm}(\langle\alpha\rangle\varphi) = 1 + \mathrm{pm}(\varphi)$.

Proposição. *Se $x \sim_n y$ e $\mathrm{pm}(\varphi) \leq n$ então $x \models \varphi$ se e só se $y \models \varphi$.*

Prova. Exercício. ∎

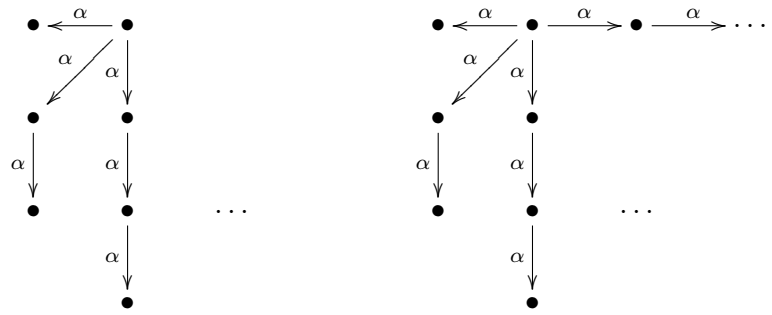
Corolário. *Se $x \sim_\omega y$ então $x \sim_{\mathrm{HML}} y$.*

Temos portanto $\sim \subseteq \sim_\omega \subseteq \sim_{\mathrm{HML}}$, e portanto as três relações coincidem em sistemas de imagens pseudo-fechadas.

Exercício. Faça uma prova directa de $\sim = \sim_\omega$ para sistemas de imagens finitas. Mostre também que para tais sistemas de tem $x \sim_n y$ se e só se x e y satisfazem exactamente as mesmas fórmulas de profundidade modal menor ou igual a n.

Teorema. *Existem sistemas para os quais $\sim \neq \sim_{\mathrm{HML}}$.*

Prova. Considere-se o st $\langle P, \rightarrow \rangle$ representado na figura seguinte:



Este sistema consiste em duas cópias dum sistema com um número infinito de ramos finitos, tendo-se numa das cópias acrescentado um ramo infinito. Mais precisamente, os estados são 0, 1, $\langle 0, m, n \rangle$ e $\langle 1, m, n \rangle$, com $n \in \omega$ e $m \leq n$, ou $\langle 1, n \rangle$ com $n \in \omega$. Há as seguintes transições:

- $0 \xrightarrow{\alpha} \langle 0, 0, n \rangle$
- $1 \xrightarrow{\alpha} \langle 1, 0, n \rangle$
- $1 \xrightarrow{\alpha} \langle 1, 0 \rangle$
- $\langle 0, m, n \rangle \xrightarrow{\alpha} \langle 0, m+1, n \rangle$ se $m < n$
- $\langle 1, m, n \rangle \xrightarrow{\alpha} \langle 1, m+1, n \rangle$ se $m < n$
- $\langle 1, n \rangle \xrightarrow{\alpha} \langle 1, n+1 \rangle$

É simples ver que os estados 0 e 1 não são bissimilares. Vamos agora ver que $0 \sim_\omega 1$, e que portanto 0 e 1 satisfazem as mesmas fórmulas. Para tal verificamos que $0 \sim_n 1$ para qualquer $n \in \omega$. O caso $n = 0$ é trivial. Para os outros casos, é imediato que qualquer transição a partir de 0 pode ser imitada por uma transição a partir de 1 para um estado bissimilar. Do mesmo modo, qualquer transição a partir de 1 para um estado da forma $\langle 1, m, n \rangle$ pode ser imitada por uma transição a partir de 0 para o estado $\langle 0, m, n \rangle$, que é bissimilar a $\langle 1, m, n \rangle$. O único caso não trivial é a transição $1 \xrightarrow{\alpha} \langle 1, 0 \rangle$, que não pode ser imitada por 0 para nenhum estado bissimilar. No entanto, é simples ver que $\langle 1, 0 \rangle \sim_n \langle 0, 0, n \rangle$ para qualquer $n \in \omega$ (verifique), e portanto para qualquer $n \in \omega$ há sempre uma transição de 0 para um estado x tal que $x \sim_n \langle 1, 0 \rangle$, pelo que resulta $0 \sim_{n+1} 1$. ∎

Capítulo 3

Exercícios

3.1 Reticulados

3.1.1

Mostre que se $f : X \to Y$ é uma função monótona entre dois reticulados completos se tem, para qualquer $S \subseteq X$,

$$\bigvee f(S) \leq f(\bigvee S) \; .$$

3.1.2

Mostre que uma função entre dois reticulados é monótona se e só se para quaisquer dois elementos x e y do domínio

$$f(x) \vee f(y) \leq f(x \vee y) \; .$$

3.1.3

Considere a lógica proposicional definida sobre um certo conjunto de símbolos proposicionais, e para cada fórmula φ defina o conjunto

$$[\![\varphi]\!] \stackrel{\text{def}}{=} \{\text{valorações que satisfazem } \varphi\} \; .$$

Mostre que se tem

1. $[\![\varphi \vee \psi]\!] = [\![\varphi]\!] \cup [\![\psi]\!]$
2. $[\![\varphi \wedge \psi]\!] = [\![\varphi]\!] \cap [\![\psi]\!]$

3.1.4

Mostre que num reticulado $\mathcal{L}$ as operações de supremo e ínfimo

$$\vee : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$$
$$\wedge : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$$

satisfazem as seguintes propriedades:

1. $x \vee (y \vee z) = (x \vee y) \vee z$ (associatividade)
2. $x \vee x = x$ (idempotência)
3. $x \vee y = y \vee x$ (comutatividade)
4. $x \wedge (y \wedge z) = (x \wedge y) \wedge z)$ (associatividade)
5. $x \wedge x = x$ (idempotência)
6. $x \wedge y = y \wedge x$ (comutatividade)
7. $x \wedge (x \vee y) = x$ (absorção)
8. $x \vee (x \wedge y) = x$ (absorção)

Mostre também que em qualquer conjunto com duas operações $\wedge$ e $\vee$ com as propriedades acima é um reticulado, se definirmos a relação de ordem por $x \leq y \overset{\text{def}}{\iff} x \vee y = y$.

3.2 Simulações, bissimulações, etc.

3.2.1

1. Mostre que $\sqsubseteq_{\mathrm{S}}$ é uma pré-ordem (i.e., uma relação reflexiva e transitiva).
2. Definindo $\sim_{\mathrm{S}}$ como

   $$x \sim_{\mathrm{S}} y \overset{\text{def}}{\iff} x \sqsubseteq_{\mathrm{S}} y \text{ e } y \sqsubseteq_{\mathrm{S}} x$$

   mostre que $\sim \subsetneq \sim_{\mathrm{S}} \subsetneq \sim_{\mathrm{T}}$.

3.2.2

Seja $\langle P, \rightarrow \rangle$ um st e defina a seguinte família de relações binárias sobre P, indutivamente em $n \in \omega$:

$$\begin{array}{rcl} \sim_0 & \stackrel{\text{def}}{=} & P \times P \\ x \sim_{n+1} y & \stackrel{\text{def}}{\iff} & \left\{ \begin{array}{l} \forall_\alpha \forall_{x'} (x \stackrel{\alpha}{\rightarrow} x' \Rightarrow \exists_{y'} (y \stackrel{\alpha}{\rightarrow} y' \text{ e } x' \sim_n y')) \\ \forall_\alpha \forall_{y'} (y \stackrel{\alpha}{\rightarrow} y' \Rightarrow \exists_{x'} (x \stackrel{\alpha}{\rightarrow} x' \text{ e } x' \sim_n y')) \end{array} \right. \end{array}$$

Defina também a relação $\sim_\omega \stackrel{\text{def}}{=} \bigcap_{n \in \omega} \sim_n$.

1. Mostre que $\sim_{n+1} \subseteq \sim_n$ para qualquer $n \in \omega$.
2. Mostre que $\sim \subseteq \sim_\omega$.
3. Mostre que se o st for de imagens finitas (i.e., os conjuntos $x \cdot \alpha \stackrel{\text{def}}{=} \{y \in P \mid x \stackrel{\alpha}{\rightarrow} y\}$ são finitos para todos os valores de $x \in P$) então $\sim = \sim_\omega$. [Sugestão: mostre que $\sim_\omega$ é uma bissimulação.]

3.2.3

Dizemos que uma relação binária $S \subseteq P \times Q$ é uma *R-simulação* entre sts $\langle P, \rightarrow \rangle$ e $\langle Q, \rightarrow \rangle$ se para quaisquer estados $x \in P$ e $y \in Q$ se tem

$$x \; S \; y \Rightarrow \left\{ \begin{array}{l} \forall_\alpha \forall_{x'} (x \stackrel{\alpha}{\rightarrow} x' \Rightarrow \exists_{y'} (y \stackrel{\alpha}{\rightarrow} y' \text{ e } x' \; R \; y')) \\ \forall_\alpha (x \stackrel{\alpha}{\not\rightarrow} \Rightarrow y \stackrel{\alpha}{\not\rightarrow}) \end{array} \right.$$

1. Mostre que dados sts arbitrários $\langle P, \rightarrow \rangle$ e $\langle Q, \rightarrow \rangle$ existe uma R-simulação $\sqsubseteq_{\text{RS}}$ que contém todas as outras R-simulações entre $\langle P, \rightarrow \rangle$ e $\langle Q, \rightarrow \rangle$, e tal que para quaisquer $x \in P$ e $y \in Q$ se tem

$$x \sqsubseteq_{\text{RS}} y \iff \left\{ \begin{array}{l} \forall_\alpha \forall_{x'} (x \stackrel{\alpha}{\rightarrow} x' \Rightarrow \exists_{y'} (y \stackrel{\alpha}{\rightarrow} y' \text{ e } x' \sqsubseteq_{\text{RS}} y')) \\ \forall_\alpha (x \stackrel{\alpha}{\not\rightarrow} \Rightarrow y \stackrel{\alpha}{\not\rightarrow}) \end{array} \right.$$

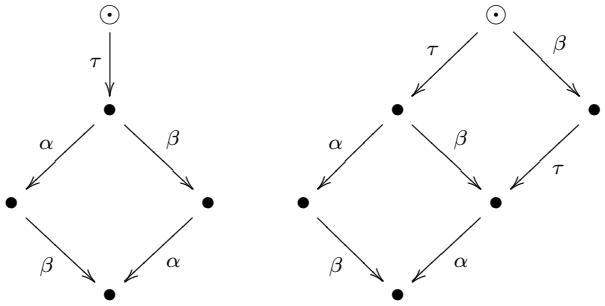
2. Mostre que $\sqsubseteq_{\text{RS}}$ é uma pré-ordem.
3. Mostre que $\sim \subsetneq \sqsubseteq_{\text{RS}} \subsetneq \sqsubseteq_{\text{S}} \subsetneq \sqsubseteq_{\text{T}}$.

3.2.4

Um st diz-se *determinista* se para quaisquer estados x, y e z e qualquer acção α se tem $y = z$ sempre que $x \stackrel{\alpha}{\rightarrow} y$ e $x \stackrel{\alpha}{\rightarrow} z$. Mostre que dois estados dum st determinista são bissimilares sse têm os mesmos traços.

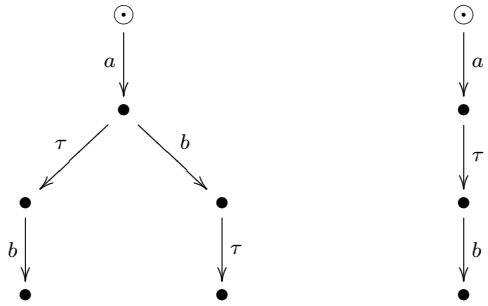
3.2.5

Mostre que os (estados iniciais dos) stas seguintes são fracamente bissimilares mas não fortemente bissimilares:



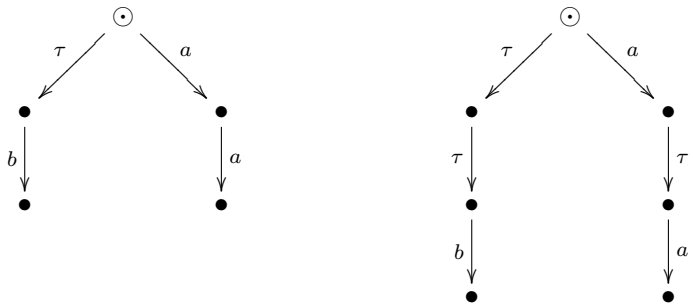
3.2.6

Mostre que os (estados iniciais dos) stas seguintes são observacionalmente congruentes mas não fortemente bissimilares:



3.2.7

Mostre que os stas seguintes são observacionalmente congruentes.



3.2.8

Verifique se os stas seguintes são observacionalmente congruentes.

3.2.9

Diz-se que uma relação $R \subseteq \mathcal{P}(\mathcal{K}) \times \mathcal{P}(\mathcal{K})$ é uma *bissimulação ramificada* se para quaisquer agentes P e Q e qualquer acção $\alpha \in Act$ se tem

$$P\ R\ Q \Rightarrow \begin{cases} \forall_{P'}(P \xrightarrow{\alpha} P' \Rightarrow ((\alpha = \tau\ \wedge\ P'\ R\ Q) \vee (\exists_{Q',Q_0,Q_0'}(Q \stackrel{\varepsilon}{\Rightarrow} Q_0 \xrightarrow{\alpha} Q_0' \stackrel{\varepsilon}{\Rightarrow} Q' \\ \qquad\qquad \wedge\ P\ R\ Q_0\ \wedge\ P'\ R\ Q_0'\ \wedge\ P'\ R\ Q'))) \\ \forall_{Q'}(Q \xrightarrow{\alpha} Q' \Rightarrow ((\alpha = \tau\ \wedge\ P\ R\ Q') \vee (\exists_{P',P_0,P_0'}(P \stackrel{\varepsilon}{\Rightarrow} P_0 \xrightarrow{\alpha} P_0' \stackrel{\varepsilon}{\Rightarrow} P' \\ \qquad\qquad \wedge\ P_0\ R\ Q\ \wedge\ P_0'\ R\ Q'\ \wedge\ P'\ R\ Q'))) . \end{cases}$$

1. Justifique que existe uma bissimulação ramificada $\approx_{\mathrm{r}}$ que contém todas as outras.

2. Mostre que $R \subseteq \mathcal{P}(\mathcal{K}) \times \mathcal{P}(\mathcal{K})$ é uma bissimulação fraca se e só se para quaisquer agentes P e Q e qualquer $\alpha \in Act$ se tem

$$P\ R\ Q \Rightarrow \begin{cases} \forall_{P'}(P \xrightarrow{\alpha} P' \Rightarrow ((\alpha = \tau\ \wedge\ P'\ R\ Q) \vee (\exists_{Q'}(Q \stackrel{\alpha}{\Rightarrow} Q'\ \wedge\ P'\ R\ Q'))) \\ \forall_{Q'}(Q \xrightarrow{\alpha} Q' \Rightarrow ((\alpha = \tau\ \wedge\ P\ R\ Q') \vee (\exists_{P'}(P \stackrel{\alpha}{\Rightarrow} P'\ \wedge\ P'\ R\ Q'))) . \end{cases}$$

   [Sugestão: comece por justificar que $P \stackrel{\widehat{\alpha}}{\Rightarrow} P'$ é equivalente à condição

$$((\alpha = \tau) \wedge (P = P')) \vee (P \stackrel{\alpha}{\Rightarrow} P') \ .\ ]$$

3. Mostre que $\approx_{\mathrm{r}} \subsetneq \approx$.

3.2.10

1. Prove $P_0 \approx Q_0 \iff (A) \iff (B)$, onde (A) e (B) são as proposições seguintes:

(A) Existe uma relação R tal que $P_0 \; R \; Q_0$ e tal que

$$P \; R \; Q \Rightarrow \left\{ \begin{array}{l} P \stackrel{\alpha}{\Rightarrow} P' \Rightarrow \exists_{Q'}(Q \stackrel{\hat{\alpha}}{\Rightarrow} Q' \; \wedge \; P' \approx R \approx Q') \\ Q \stackrel{\alpha}{\Rightarrow} Q' \Rightarrow \exists_{P'}(P \stackrel{\hat{\alpha}}{\Rightarrow} P' \; \wedge \; P' \approx R \approx Q') \end{array} \right.$$

(B) Existe uma relação R tal que $P_0 \; R \; Q_0$ e tal que

$$P \; R \; Q \Rightarrow \left\{ \begin{array}{l} P \stackrel{\alpha}{\rightarrow} P' \Rightarrow \exists_{Q'}(Q \stackrel{\hat{\alpha}}{\Rightarrow} Q' \; \wedge \; P' \sim R \approx Q') \\ Q \stackrel{\alpha}{\rightarrow} Q' \Rightarrow \exists_{P'}(P \stackrel{\hat{\alpha}}{\Rightarrow} P' \; \wedge \; P' \approx R \sim Q') \end{array} \right.$$

2. Mostre que pode ter-se $P_0 \; R \; Q_0$ para uma relação R tal que

$$P \; R \; Q \Rightarrow \left\{ \begin{array}{l} P \stackrel{\alpha}{\rightarrow} P' \Rightarrow \exists_{Q'}(Q \stackrel{\hat{\alpha}}{\Rightarrow} Q' \; \wedge \; P' \approx R \approx Q') \\ Q \stackrel{\alpha}{\rightarrow} Q' \Rightarrow \exists_{P'}(P \stackrel{\hat{\alpha}}{\Rightarrow} P' \; \wedge \; P' \approx R \approx Q') \end{array} \right.$$

e ter-se no entanto $P_0 \not\approx Q_0$ (sugestão: considere os agentes $\tau.a.\mathbf{0}$ e $\mathbf{0}$).

3.3 Álgebra de Processos

3.3.1

Represente graficamente os stas de

1. $(a + b) \parallel (c + d)$
2. $((a + b) \parallel (c + d)) \parallel e$
3. $aaa \parallel (bbb \parallel ccc)$
4. $a \parallel (b \parallel (c \parallel d))$

3.3.2

Represente graficamente os stas de

1. $\langle A, \{A \stackrel{\text{def}}{=} a.b.A + c\}\rangle$
2. $\langle d.A, \Sigma\rangle$, onde Σ contém as equações

$$\begin{array}{rcl} A & \stackrel{\text{def}}{=} & a.b.B + c.A \\ B & \stackrel{\text{def}}{=} & b.A \end{array}$$

3. $\langle A \parallel B, \Sigma \rangle$, onde Σ contém as equações

$$\begin{array}{rcl} A & \stackrel{\text{def}}{=} & a.A \\ B & \stackrel{\text{def}}{=} & b.B \end{array}$$

4. $\langle A \parallel B, \Sigma \rangle$, onde Σ contém as equações

$$\begin{array}{rcl} A & \stackrel{\text{def}}{=} & a.B \\ B & \stackrel{\text{def}}{=} & b.A \end{array}$$

3.3.3

Escreva um agente elementar que represente uma máquina de vender chocolates cujo comportamento é descrito informalmente como se segue:

1. A máquina pode receber moedas de 100$ ou 200$, e pode entregar chocolates grandes ou pequenos.
2. Para receber um chocolate o utilizador deve carregar na tecla "pequeno"ou "grande", consoante o tipo de chocolate que deseja.
3. A máquina entrega um chocolate pequeno apenas se tiver recebido pelo menos 100$.
4. A máquina entrega um chocolate grande apenas se tiver recebido 200$.
5. A máquina pode receber no máximo 200$ (numa ou em duas moedas).
6. Quando recebe uma moeda ou entrega um chocolate a máquina actualiza o seu saldo correspondentemente.

3.3.4

Escreva um agente elementar que represente um contador com dois botões, *inc* e *dec*. O contador tem um valor mínimo de zero; sempre que se prime *inc* o valor é incrementado de uma unidade e quando se prime *dec* o valor é decrementado de uma unidade, o que só é possível se o valor não for zero.

3.3.5

Represente graficamente parte do sta de C, com a equação

$$C \stackrel{\text{def}}{=} inc.(C \parallel dec) \; .$$

3.3.6

Seja A uma álgebra com operações binárias $+$, $\between$, uma constante $\mathbf{0}$ e operações unárias $\alpha \in Act$, satisfazendo as seguintes equações:

1. $x + (y + z) = (x + y) + z$,
2. $x + \mathbf{0} = x$,
3. $x + y = y + x$,
4. $x + x = x$,
5. $(x \between y) \between z = x \between (y \between z + z \between y)$,
6. $\alpha x \between y = \alpha(x \between y + y \between x)$,
7. $(x + y) \between z = x \between z + y \between z$,
8. $\mathbf{0} \between x = \mathbf{0}$,
9. $x \between \mathbf{0} = x$.

Mostre que A é uma álgebra de intercalação se definirmos

$$x \parallel y = x \between y + y \between x \ .$$

3.3.7

Seja $\mathcal{P}_\ell(\mathcal{K})$ a linguagem definida indutivamente por:

- $\mathbf{0} \in \mathcal{P}_\ell(\mathcal{K})$;
- $\mathcal{K} \subseteq \mathcal{P}_\ell(\mathcal{K})$;
- se $P \in \mathcal{P}_\ell(\mathcal{K})$ e $\alpha \in Act$ então $\alpha.P \in \mathcal{P}_\ell(\mathcal{K})$;
- se $P, Q \in \mathcal{P}_\ell(\mathcal{K})$ então $(P + Q), (P \parallel Q), (P \between Q) \in \mathcal{P}_\ell(\mathcal{K})$.

Usaremos as convenções habituais de omissão de parênteses e convencionamos que $P + Q \between R$ é lido como $P + (Q \between R)$ e $\alpha.P \between Q$ é lido como $(\alpha.P) \between Q$.

Seja ainda $\rightarrow \subseteq \mathcal{P}_\ell(\mathcal{K}) \times Act \times \mathcal{P}_\ell(\mathcal{K})$ uma relação de transição definida pelas regras habituais, mais a seguinte:

$$\frac{P \xrightarrow{\alpha} P'}{P \between Q \xrightarrow{\alpha} P' \parallel Q} \ .$$

Designamos a operação $\between$ por *composição paralela à esquerda*, ou simplesmente *composição à esquerda*.

1. Prove que para quaisquer agentes $P, Q, R \in \mathcal{P}_\ell(\mathcal{K})$ se tem:
   - $P \parallel Q \sim P \between Q + Q \between P$;
   - $(P \between Q) \between R \sim P \between (Q \parallel R)$;
   - $\alpha.P \between Q \sim \alpha.(P \parallel Q)$;
   - $(P + Q) \between R \sim P \between R + Q \between R$;
   - $\mathbf{0} \between P \sim \mathbf{0}$;
   - $P \between \mathbf{0} \sim P$.
2. Mostre que a relação de bissimilaridade sobre $\mathcal{P}_\ell(\mathcal{K})$ é uma congruência para a operação $\between$ (e verifique que o continua a ser para as restantes operações).

3.3.8

Considere a linguagem $\mathcal{P}(\mathcal{K})$ enriquecida com o operador binário $\lfloor$ e o sistema de transição enriquecido com as regras seguintes:

$$\mathbf{Lco}_1 \quad \frac{E \xrightarrow{\alpha} E'}{E \lfloor F \xrightarrow{\alpha} E' \mid F} \qquad\qquad \mathbf{Lco}_2 \quad \frac{E \xrightarrow{\ell} E' \quad F \xrightarrow{\overline{\ell}} F'}{E \lfloor F \xrightarrow{\tau} E' \mid F'}$$

1. Mostre que se tem $(P+Q)\lfloor R \sim (P\lfloor R)+(Q\lfloor R)$, para quaisquer agentes P, Q e R.
2. Dê um exemplo de agentes P, Q e R para os quais $(P + Q) \mid R \not\sim (P \mid R) + (Q \mid R)$.

3.3.9

1. Mostre que $\sim$ é congruência para $|$, $\backslash L$ e $[f]$, onde $L \subseteq \mathcal{L}$ e f é uma função de re-etiquetação.
2. Prove que a álgebra $\mathcal{P}(\mathcal{K})/\sim$ satisfaz as seguintes propriedades:
   (a) $x \mid (y \mid z) = (x \mid y) \mid z$
   (b) $x \mid \mathbf{0} = x$
   (c) $x \mid y = y \mid x$
   (d) $(\sum_{i=1}^{n} \alpha_i.x_i) \mid (\sum_{j=1}^{m} \beta_j.y_j) =$

$$= \sum_{i=1}^{n} \alpha_i.(x_i \mid (\sum_{j=1}^{m} \beta_j.y_j)) + \sum_{j=1}^{m} \beta_j.((\sum_{i=1}^{n} \alpha_i.xi) \mid y_j) + \sum_{\alpha_i = \overline{\beta_j} \neq \tau} \tau.(x_i \mid y_j)$$

(e) $x\backslash\emptyset = x$

(f) $x\backslash K\backslash L = x\backslash(K \cup L)$

(g) $\mathbf{0}\backslash L = \mathbf{0}$

(h) $(x + y)\backslash L = x\backslash L + y\backslash L$

(i) $(\alpha.x)\backslash L = \mathbf{0}$ se $\alpha \in L$ ou $\overline{\alpha} \in L$

(j) $(\alpha.x)\backslash L = \alpha.x\backslash L$ se $\alpha \notin L$ e $\overline{\alpha} \notin L$

(k) $x[\mathrm{id}] = x$

(l) $x[f][g] = x[f; g]$

(m) $\mathbf{0}[f] = \mathbf{0}$

(n) $(x + y)[f] = x[f] + y[f]$

(o) $(x \mid y)[f] = x[f] \mid y[f]$ se f for bijectiva

(p) $(\alpha.x)[f] = f(\alpha).x[f]$

(q) $(x[f])\backslash L = (x\backslash f^{-1}(L))[f]$

(r) $(x\backslash L)[f] = (x[f])\backslash f(L)$ se f for bijectiva

3. Mostre que a equação (o) não é verdadeira em geral se f não for bijectiva, mas dê também um exemplo de agentes P e Q e de uma função de re-etiquetação não bijectiva f tais que $(P \mid Q)[f] \sim P[f] \mid Q[f]$.

3.3.10

Considere as seguintes definições:

$$\begin{array}{llll} S \stackrel{\text{def}}{=} s \cdot w \cdot S & & C \stackrel{\text{def}}{=} D + E \\ A \stackrel{\text{def}}{=} \overline{s}.a.\overline{w}.A & & D \stackrel{\text{def}}{=} b.C \\ B \stackrel{\text{def}}{=} a.B + b.B & & E \stackrel{\text{def}}{=} a.C \end{array}$$

1. Represente graficamente os stas, dados pela semântica operacional estrutural, cujos estados iniciais são (i) S, (ii) A, (iii) B, (iv) $A[b/a]$, (v) $A \mid S$, e (vi) $((A \mid S) \mid A[b/a])\backslash\{s, w\}$, respectivamente.

2. Prove $(A \mid S \mid A[b/a])\backslash\{s, w\} \not\approx B$. Ter-se-á $(A \mid S \mid A[b/a])\backslash\{s, w\} \sim B$?

3. Prove $B \sim C$, sem construir nenhuma bissimulação. [Sugestão: comece por provar $C \sim b.C + a.C$.]

3.3.11

Considere as seguintes definições:

$$\begin{array}{lcllcl} S & \stackrel{\text{def}}{=} & s \cdot w \cdot S & C & \stackrel{\text{def}}{=} & \tau.E + \tau.D \\ A & \stackrel{\text{def}}{=} & \overline{s}.a.b.\overline{w}.A & D & \stackrel{\text{def}}{=} & a.a.C \\ B & \stackrel{\text{def}}{=} & \tau.a.a.B + \tau.b.b.B & E & \stackrel{\text{def}}{=} & b.b.C \end{array}$$

1. Represente graficamente os stas, dados pela semântica operacional estrutural, cujos estados iniciais são B, $A[a/b]$ e $((A[a/b] \mid S) \mid A[b/a])\backslash\{s, w\}$, respectivamente.

2. (a) Prove $B \approx (A[a/b] \mid S \mid A[b/a])\backslash\{s, w\}$ por meio duma bissimulação fraca.
   (b) Prove $B \simeq (A[a/b] \mid S \mid A[b/a])\backslash\{s, w\}$ algebricamente.
   (c) Mostre que se $P \stackrel{\tau}{\not\to}$ então $P \not\approx (A[a/b] \mid S \mid A[b/a])\backslash\{s, w\}$.

3. Prove $B \sim C$, sem construir nenhuma bissimulação. [Sugestão: comece por provar $C \sim \tau.b.b.C + \tau.a.a.C$.]

3.3.12

Considere as seguintes definições:

$$\begin{array}{lcl} T & \stackrel{\text{def}}{=} & \overline{\texttt{tic}}.T \\ D & \stackrel{\text{def}}{=} & \texttt{com.com.com.com.}\overline{\texttt{tic}}.D \end{array}$$

1. Represente graficamente os stas, dados pela semântica operacional estrutural, cujos estados iniciais são T, D e $(T[\texttt{com/tic}] \mid D)\backslash\texttt{com}$, respectivamente.

2. Prove $T \approx (T[\texttt{com/tic}] \mid D)\backslash\texttt{com}$ por meio duma bissimulação. Ter-se-á também $T \simeq (T[\texttt{com/tic}] \mid D)\backslash\texttt{com}$?

3. Prove $T \approx (T[\texttt{com/tic}] \mid D)\backslash\texttt{com}$ algebricamente (Sugestão: mostre $\tau.T \approx (T[\texttt{com/tic}] \mid D)\backslash\texttt{com}$).

3.3.13

Prove as seguintes asserções, justificando algebricamente ou por meio de bissimulações:

1. $(a \mid (abB)[b/a])\backslash b \sim a$
2. $(a \mid (\overline{b}a)[a/b])\backslash a \approx \mathbf{0}$

3.3.14

Prove que qualquer CCS-álgebra satisfaz a lei

$$x + \tau.(x + y) = \tau.(x + y) \ .$$

3.3.15

Prove as seguintes asserções, justificando algebricamente ou por meio de bissimulações:

1. $P \simeq Q$, sabendo que $P \simeq a(\tau b + \tau P) + ab$ e $Q \simeq a(\tau b + \tau Q) + a\tau Q$
2. $P \simeq Q$, sabendo que $P \simeq \tau aP + (a\tau \,|\, b)\backslash b$ e $Q \simeq (a \,|\, bQ)\backslash b + a\tau Q + \tau\tau a\tau Q$

3.3.16

Considere as seguintes famílias de agentes, onde X é um conjunto qualquer, $s \in X^\omega$ e $n \in \omega$, e se utiliza a notação $s[x/n]$ para o *array* que resulta de escrever $x \in X$ em s na posição n (i.e., $s[x/n](n) = x$, e se $m \neq n$ então $s[x/n](m) = s(m)$):

$$\begin{aligned}
Array_s &\stackrel{\text{def}}{=} \sum_{x,n} \mathsf{write}_{x,n}.Array_{s[x/n]} + \sum_n \mathsf{choose}_n.\overline{\mathsf{read}}_{s(n)}.Array_s \\
Point_0 &\stackrel{\text{def}}{=} \mathsf{inc}.Point_1 + \overline{\mathsf{pos}}_0.Point_0 \\
Point_{n+1} &\stackrel{\text{def}}{=} \mathsf{inc}.Point_{n+2} + \mathsf{dec}.Point_n + \overline{\mathsf{pos}}_{n+1}.Point_{n+1} \\
StackControl &\stackrel{\text{def}}{=} \sum_x \mathsf{push}_x.\sum_n \mathsf{pos}_n.\overline{\mathsf{inc}}.\overline{\mathsf{write}}_{x,n}.StackControl \\
&\quad + \mathsf{pop}.\overline{\mathsf{dec}}.\sum_n \mathsf{pos}_n.\overline{\mathsf{choose}}_n.\sum_x \mathsf{read}_x.\overline{\mathsf{top}}_x.StackControl \\
Stack_{s,n} &\equiv (StackControl \,|\, (Array_s \,|\, Point_n))\backslash L \\
L &= \{\mathsf{inc},\ \mathsf{dec},\ \mathsf{pos}_n,\ \mathsf{write}_{x,n},\ \mathsf{read}_x,\ \mathsf{choose}_n \mid n \in \omega,\ x \in X\}
\end{aligned}$$

1. Mostre algebricamente que se $n > 0$ então

$$Stack_{s,n} \simeq \sum_x \mathsf{push}_x.Stack_{s[x/n],n+1} + \mathsf{pop}.\overline{\mathsf{top}}_{s(n-1)}.Stack_{s,n-1} \ .$$

[Nota: sempre que achar oportuno pode, para simplificar os cálculos, aplicar em conjunto a lei de expansão, a distributividade da restrição sobre a soma e as leis que relacionam a restrição com a prefixação.]

2. Reescreva o agente *StackControl* usando o cálculo de passagem de valores.

3.3.17

Sejam C e Seq agentes tais que

$$\begin{aligned} C &\simeq \mathsf{in.a.}\overline{\mathsf{out}}.C \\ Seq &\approx a_1.a_2.Seq \end{aligned}$$

Mostre algebricamente que $Seq \approx (\overline{c_1} \,|\, C[f_1] \,|\, C[f_2]) \backslash \{c_1, c_2\}$, onde as funções de re-etiquetação f_1 e f_2 são definidas por

f_1			f_2		
a	$\mapsto$	a_1	a	$\mapsto$	a_2
in	$\mapsto$	c_1	in	$\mapsto$	c_2
out	$\mapsto$	c_2	out	$\mapsto$	c_1

3.3.18

Altere a especificação do ABP feita nas aulas teóricas, de modo a que o receptor deixe de ter um temporizador, em vez disso enviando um $\overline{\mathsf{reply}}_b$ imediatamente após $\overline{\mathsf{deliver}}$ e, depois disso, sempre que receber um trans_b; se após $\overline{\mathsf{deliver}}$ ou trans_b o receptor receber $\mathsf{trans}_{\hat{b}}$ então volta a fazer $\overline{\mathsf{deliver}}$. Mostre que o novo ABP é (fracamente) bissimilar ao agente B definido por $B \stackrel{\text{def}}{=} \mathsf{accept.deliver}.B$. [Sugestão: verifique se a bissimulação das aulas teóricas ainda é uma bissimulação neste caso.]

3.3.19

1. Prove as seguintes asserções, justificando algebricamente:

   (a) $P \sim Q$, sabendo que $P \sim a(b+P)+ab$ e $Q \sim a(b+Q)+ab+ab$

   (b) $P \sim Q$, sabendo que $P \sim aP + (a \,|\, b)\backslash b$ e $Q \sim (a \,|\, bQ)\backslash b + aQ$

2. Para cada um dos casos anteriores defina uma bissimulação R a menos de $\sim$ tal que $P \; R \; Q$.

3.3.20

Sem recorrer ao teorema da unicidade de soluções de equações recursivas módulo congruência observacional, prove que

1. $P \simeq P'$ sabendo que $P \simeq aP$ e $P' \simeq aP'$
2. $P \simeq P'$ e $Q \simeq Q'$ sabendo que $P \simeq aQ + b$, $P' \simeq aQ' + b$, $Q \simeq bP$ e $Q' \simeq bP'$.

3.3.21

Recorrendo ao teorema da unicidade de soluções de equações recursivas módulo congruência observacional, mostre que $P \simeq Q$, sabendo que $P \simeq aQ$ e $Q \simeq aP$.

3.3.22

Mostre que os agentes A e B são observacionalmente congruentes:

$$\begin{aligned} A &\stackrel{\text{def}}{=} \tau.\alpha.A + (\alpha.\tau \mid \beta)\backslash\beta \\ B &\stackrel{\text{def}}{=} \alpha + \alpha.B + \tau.\alpha.B \end{aligned}$$

3.3.23

Mostre que os agentes A e B são observacionalmente congruentes:

$$\begin{aligned} A &\stackrel{\text{def}}{=} \tau.\alpha.A + (\alpha.\tau \mid)\backslash\beta \\ B &\stackrel{\text{def}}{=} (\alpha \mid \beta.B)\backslash\beta + \alpha.\tau.B + \tau.\tau.\alpha.\tau.B \end{aligned}$$

3.3.24

Mostre que os agentes A e B são observacionalmente congruentes:

$$\begin{aligned} A &\stackrel{\text{def}}{=} \tau.(\alpha.\tau.A + \beta) \\ B &\stackrel{\text{def}}{=} \alpha.\tau.B + \tau.(\alpha.B + \beta) \end{aligned}$$

3.3.25

Mostre que os agentes A e B são observacionalmente congruentes:

$$\begin{aligned} A &\stackrel{\text{def}}{=} \alpha.(\tau.\beta + \tau.A) + \alpha.\beta \\ B &\stackrel{\text{def}}{=} \alpha.(\tau.\beta + \tau.B) + \alpha.\tau.B \end{aligned}$$

3.3.26

Desenhe os stas de:

1. $\mathrm{rec}X.(\alpha.X + \beta)$
2. $\mathrm{rec}X.(\alpha.(\tau.X + \beta.X))$

3.3.27

Mostre que, sendo A e B os agentes definidos por

$$\begin{aligned} A &\stackrel{\text{def}}{=} a.A + b.B \\ B &\stackrel{\text{def}}{=} c.A + d.B \end{aligned}$$

se tem $A \simeq \mathrm{rec}X.(a.X + b.\mathrm{rec}Y.(c.X + d.Y))$.

3.3.28

Mostre que $A \simeq B$, onde:

$$\begin{aligned} A &\stackrel{\text{def}}{=} a.c.A + \tau.(b.d.A + \tau.A) \\ B &\stackrel{\text{def}}{=} \tau.(a.c.B + b.d.B) \end{aligned}$$

Verifique este resultado semanticamente.

3.3.29

1. Para cada um dos agentes finitos seguintes obtenha, utilizando as leis algébricas da bissimulação forte, um que lhe seja fortemente bissimilar mas no qual não surjam comunicações, restrições ou re-etiquetações.
   (a) $(ab \,|\, cd)\backslash d$
   (b) $((ab \,|\, cd)\backslash d)[a/d]$
   (c) $((\overline{a}b \,|\, ac \,|\, \overline{a}d)[e/a])\backslash e$
   (d) $((ac \,|\, bc)[b/a])\backslash a$
2. Obtenha, novamente pelas leis algébricas da bissimulação forte, formas normais para cada um dos agentes anteriores.

3.3.30

Utilizando as leis algébricas da congruência observacional converta para forma normal total os seguintes agentes finitos:

1. $\alpha.\tau(\tau.\beta + \tau.\gamma)$
2. $\tau.(\alpha.\tau + \beta)$
3. $(\alpha.\beta \,|\, \tau.\tau)\backslash\beta$

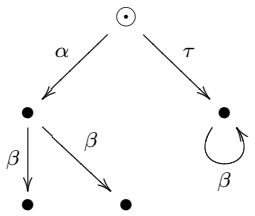
3.4 Lógica de Hennessy e Milner (HML)

3.4.1

Diga quais das seguintes fórmulas,

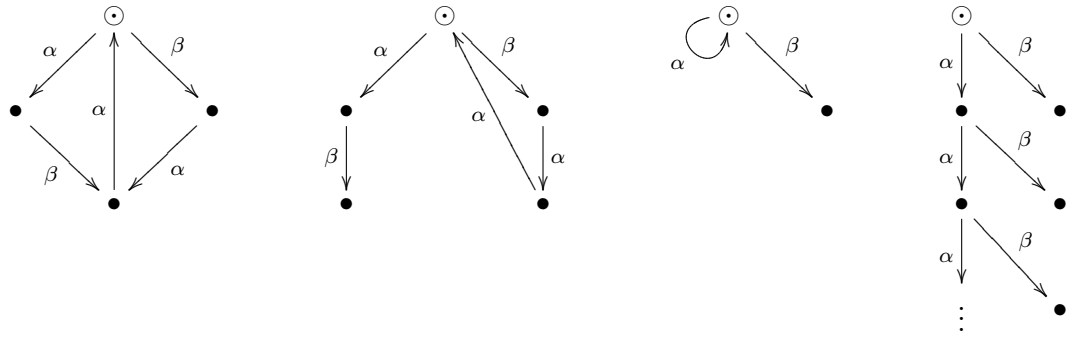
1. $\langle\alpha\rangle\top$
2. $\langle\alpha\rangle\bot$
3. $[\alpha]\top$
4. $[\alpha]\bot$
5. $[\beta]\bot$
6. $\langle\alpha\rangle[\beta]\langle\tau\rangle\top$
7. $\langle\alpha\rangle\langle\beta\rangle\langle\tau\rangle\top$
8. $\langle\alpha\rangle[\beta]\bot$
9. $\langle\alpha\rangle([\beta][\alpha]\bot \wedge \langle\beta\rangle\top)$
10. $[\tau](\langle\beta\rangle\top \wedge [\beta]\langle\beta\rangle\top)$,

são verdadeiras no estado inicial do sta:



3.4.2

Para cada par dos stas seguintes obtenha, ou justifique que tal não é possível, uma fórmula HML satisfeita por apenas um deles:



3.4.3

Mostre que $\alpha(\beta(\gamma+\alpha)+\gamma) \models \langle\alpha\rangle([\beta](\langle\gamma\rangle\top \wedge \langle\alpha\rangle\top) \wedge \langle\beta\rangle\top \wedge \langle\gamma\rangle\top)$, assumindo que $\alpha \neq \beta$, $\alpha \neq \gamma$ e $\beta \neq \gamma$. Mostre também que este resultado já não se verifica se $\beta = \gamma$.

3.4.4

Duas fórmulas φ e ψ são *equivalentes*, e escrevemos $\varphi \equiv \psi$, se para qualquer estado p de qualquer st se tem $p \models \varphi \Leftrightarrow p \models \psi$. Prove as seguintes afirmações:

1. $\bot \equiv \langle\alpha\rangle\bot$
2. $\langle\alpha\rangle(\varphi \vee \psi) \equiv \langle\alpha\rangle\varphi \vee \langle\alpha\rangle\psi$
3. $[\alpha]\top \equiv \top$
4. $[\alpha](\varphi \wedge \psi) \equiv [\alpha]\varphi \wedge [\alpha]\psi$
5. $[\alpha]\varphi \wedge \langle\alpha\rangle\top \equiv [\alpha]\varphi \wedge \langle\alpha\rangle\varphi$
6. $[\alpha]\varphi \wedge [\alpha]\bot \equiv [\alpha]\bot$

Mostre ainda que a relação $\equiv$ é uma congruência para as operações $\varphi \mapsto \langle\alpha\rangle\varphi$, $\varphi \mapsto [\alpha]\varphi$, $\langle\varphi,\psi\rangle \mapsto \varphi \wedge \psi$, $\langle\varphi,\psi\rangle \mapsto \varphi \vee \psi$, $\varphi \mapsto \neg\varphi$, e que a álgebra $\mathcal{L}_{\mathrm{HML}}/\equiv$ é um reticulado cujos ínfimos são dados por $[\varphi] \wedge [\psi] = [\varphi \wedge \psi]$,

cujos supremos são dados por $[\varphi] \vee [\psi] = [\varphi \vee \psi]$, com mínimo $[\bot]$ e com máximo $[\top]$. [A esta álgebra chama-se a *álgebra de Lindenbaum* da lógica (de Hennessy e Milner, neste caso, mas construções similares se fazem para outras lógicas).]

3.4.5

Seja $\langle P, \rightarrow \rangle$ um st. Sendo um conjunto *aberto* um conjunto $X \subseteq P$ cujo complementar é fechado (i.e., $\overline{P \setminus X} = P \setminus X$), mostre que os conjuntos da forma $A(\varphi) = \{x \mid x \models \varphi\}$ são abertos. Mostre também que qualquer aberto é da forma $\bigcup_i A(\varphi_i)$ (sugestão: comece por mostrar que se X é aberto e $x \in X$ então existe φ tal que $x \models \varphi$ e $A(\varphi) \subseteq X$).

Apêndice A

Breve introdução à álgebra universal

Aqui afloramos brevemente algumas noções, muito preliminares, de álgebra universal. Referiremos apenas o caso em que há um só género.

A.1

Seja D um conjunto. Uma *operação sobre* D é uma função $f : D^n \to D$, onde n é um número natural, dito a *aridade* da operação. Operações de aridade zero são denominadas *constantes* de D.

Uma *álgebra*, $A = \langle D_A, O_A \rangle$, consiste num conjunto D_A, denominado *domínio* ou *suporte* da álgebra e num conjunto O_A de operações sobre D_A.

Nota. Por vezes abusaremos da linguagem, confundindo uma álgebra A com o seu suporte D_A.

Exemplo. Semigrupos, monóides, grupos e anéis são álgebras. As operações são as seguintes:

- Semigrupos têm apenas uma operação (binária);
- Monóides têm uma operação binária e uma constante (o elemento neutro);
- Grupos têm uma operação binária, uma constante e uma operação unária (inverso);

- Anéis têm as operações correspondentes à estrutura de grupo (zero, adição e elemento simétrico) e ainda uma outra operação binária (multiplicação); os anéis com unidade têm ainda uma outra constante (o elemento neutro da multiplicação).

Obviamente, um corpo é também uma álgebra (em particular é um anel), mas aquilo que distingue um corpo dum anel não tem carácter algébrico, pois consiste numa operação (inverso) que não está definida em todo o anel.

Exercício. O conjunto dos números naturais munido das operações de sucessor e predecessor forma uma álgebra?

A.2

Uma *assinatura*, $\Sigma = \langle Ops, \nu \rangle$, consiste num conjunto de *símbolos de operação*, Ops, e numa função ν que a cada símbolo de operação atribui um número natural, dito a *aridade* do símbolo. Também se pode dizer *tipo de similaridade* em vez de assinatura, ou *tipo operacional*.

Seja $\Sigma = \langle Ops, \nu \rangle$ uma assinatura. Uma Σ-álgebra é uma álgebra A equipada com uma função $(\cdot)_A : Ops \to O_A$, designada por *interpretação*, que a cada símbolo n-ário o de Ops faz corresponder uma operação n-ária o_A de O_A.

Exemplo. Seja Σ uma assinatura com os seguintes símbolos: **0** e **1** (0-ários), **s** (unário), **a** e **m** (binários). Qualquer anel, com a correspondência

$$\begin{aligned} \mathbf{0}_A &= 0\,, \\ \mathbf{1}_A &= 1\,, \\ \mathbf{a}_A &= +\,, \\ \mathbf{m}_A &= \cdot\,, \end{aligned}$$

é uma Σ-álgebra.

A.3

Sejam A e B duas Σ-álgebras. Um *homomorfismo* $h : A \to B$ é uma função $h : D_A \to D_B$ que "preserva"as operações, i.e., tal que, para qualquer símbolo de operação o de aridade n e $x_1, \ldots, x_n \in D_A$,

$$h(o_A(x_1, \ldots, x_n)) = o_B(h(x_1), \ldots, h(x_n))\,.$$

Um *isomorfismo* $i : A \to B$ é um homomorfismo bijectivo (cujo inverso é claramente também um homomorfismo).

Exemplo. Sejam A e B dois anéis. Uma função $h : A \to B$ é um homomorfismo de anéis sse é um homomorfismo de A e B vistos como álgebras para a assinatura Σ do Exemplo A.2.

A.4

Seja A uma álgebra. Uma relação de equivalência $\equiv$ sobre D_A diz-se uma relação de *congruência* (sobre A) se para qualquer $n \in \omega$, operação $f \in O_A$ de aridade n e $x_1, \ldots, x_n, y_1, \ldots, y_n \in D_A$,

$$((x_1 \equiv y_1) \wedge \ldots \wedge (x_n \equiv y_n)) \Rightarrow (f(x_1, \ldots, x_n) \equiv f(y_1, \ldots, y_n)) \; .$$

Seja f uma operação n-ária de A e defina-se a função $[f] : (D_A/\equiv)^n \to D_A/\equiv$, para qualquer $x_1, \ldots, x_n \in D_A$, como

$$[f]([x_1], \cdots, [x_n]) = [f(x_1, \ldots, x_n)] \; , \tag{A.1}$$

onde $[x]$ é a classe de equivalência de x, neste caso designada por *classe de congruência*. Designaremos por $O_A/\equiv$ o conjunto de todas as funções $[f]$ assim definidas, e por $A/\equiv$ a álgebra $\langle D_A/\equiv, O_A/\equiv\rangle$, dita *álgebra quociente*. Se Σ for uma assinatura e A uma Σ-álgebra, então $A/\equiv$ é também uma Σ-álgebra; a cada símbolo de operação o de Σ corresponde em $A/\equiv$ a operação $[o_A]$; isto é,

$$o_{A/\equiv} = [o_A] \; . \tag{A.2}$$

Exercício. Verifique que $[f]$ é de facto uma função.

Teorema. *Seja A uma álgebra e $\{\equiv_i\}_{i\in I}$ uma família de congruências sobre A. Então $\equiv \stackrel{def}{=} \bigcap_{i\in I} \equiv_i$ é uma congruência sobre A.*

Prova. Exercício (simples).

Seja A uma álgebra e ρ uma relação binária sobre D_A. A *congruência gerada* por ρ é a menor relação de congruência $\equiv_\rho$ que contém ρ; isto é, para a qual $x\rho y \Rightarrow x \equiv_\rho y$. Pelo teorema anterior, uma tal congruência de facto existe e é dada explicitamente por

$$\equiv_\rho \stackrel{\text{def}}{=} \bigcap \{\equiv \subseteq A \times A \mid \equiv \text{ é uma congruência e } \rho \subseteq \equiv\} \; .$$

Exercício. Dê uma definição indutiva de $\equiv_\rho$.

A.5

A relação fundamental entre congruências e homomorfismos é expressa pelo seguinte resultado.

Teorema. *Uma relação binária $\equiv$ sobre D_A é uma relação de congruência sse existe uma Σ-álgebra B e um homomorfismo $h : A \to B$ tal que $x \equiv y \iff h(x) = h(y)$ para qualquer $x, y \in D_A$.*

Prova. Seja $\equiv$ uma congruência sobre A. Então $A/\equiv$ é uma Σ-álgebra, e a função $[\cdot] : D_A \to D_A/\equiv$ que a cada $x \in D_A$ faz corresponder a classe de congruência $[x]$ é um homomorfismo: seja o um símbolo de operação n-ário; pelas equações (A.1) e (A.2) tem-se

$$[o_A(x_1, \ldots, x_n)] = o_{A/\equiv}([x_1], \ldots, [x_n]) \ .$$

Além disso tem-se $x \equiv y \iff [x] = [y]$, o que prova a existência dum homomorfismo nas condições pretendidas.

Agora seja $h : A \to B$ um homomorfismo de Σ-álgebras. Defina-se a relação binária $\equiv$ sobre D_A dada por $x \equiv y$ se $h(x) = h(y)$. É simples provar que esta relação é de equivalência. Além disso é de congruência: seja o um símbolo de operação n-ário e suponha-se $x_i \equiv y_i$ $(1 \leq i \leq n)$ em D_A. Então

$$h(o_A(x_1, \ldots, x_n)) = o_B(h(x_1), \ldots, h(x_n)) = o_B(h(y_1), \ldots, h(y_n)) = h(o_A(y_1, \ldots, y_n)) \ ,$$

ou seja, $o_A(x_1, \ldots, x_n) \equiv o_A(y_1, \ldots, y_n)$. ∎

A.6

Seja Σ uma assinatura. Um *termo* sobre Σ (abrev. Σ-termo) é um elemento do conjunto T_Σ definido indutivamente como se segue:

1. se o é um símbolo de constante (i.e., 0-ário), então $o \in T_\Sigma$;
2. sejam $t_1, \ldots, t_n$ Σ-termos e o um símbolo de operação n-ário; então $ot_1 \cdots t_n \in T_\Sigma$.

Os Σ-termos são portanto sempre *strings* de símbolos de operação.

Cada símbolo de operação n-ário define uma operação n-ária sobre T_Σ, dada por

$$o_{T_\Sigma}(t_1, \ldots, t_n) = ot_1 \cdots t_n \ ,$$

e portanto T_Σ tem também a estrutura duma Σ-álgebra, designada por *álgebra dos termos* (sobre Σ).

Teorema. *Seja Σ uma assinatura e A uma Σ-álgebra. Então existe um e um só homomorfismo $h : T_\Sigma \to A$.*

Prova. Ser um homomorfismo significa que, para quaisquer n Σ-termos t_1, $\ldots$, t_n ($n \in \omega$) e símbolo de operação n-ário o, h deve satisfazer a condição

$$h(o_{T_\Sigma}(t_1, \ldots, t_n) = o_A(h(t_1), \ldots, h(t_n)) \ ,$$

ou, equivalentemente,

$$h(ot_1 \ldots t_n) = o_A(h(t_1), \ldots, h(t_n)) \ .$$

Esta última condição é uma definição indutiva de h: se $n = 0$ então o é um símbolo de constante e $h(o) = o_A$; e a função h de facto existe e é única porque o valor $h(ot_1 \ldots t_n)$ é definido à custa de valores de h em Σ-termos cujo comprimento é estritamente inferior ao de $ot_1 \ldots t_n$. Ou seja, existe uma e uma só função que satisfaz a condição, o que termina a prova. ∎

Escreveremos habitualmente t_A para o valor $h(t)$ dum termo em A, generalizando a notação utilizada para símbolos de constante.

Esta propriedade leva a que T_Σ seja também conhecida por Σ-álgebra *inicial*, uma terminologia que provem da teoria das categorias, ou Σ-álgebra *livre*.

A.7

Seja Σ uma assinatura e G um conjunto. Um *termo sobre Σ e G*, ou *Σ-termo sobre G*, é um elemento do conjunto $T_\Sigma\langle G\rangle$, definido indutivamente como se segue:

1. $G \subseteq T_\Sigma\langle G\rangle$;
2. se o é um símbolo de constante (i.e., 0-ário), então $o \in T_\Sigma\langle G\rangle$;
3. sejam $t_1, \ldots, t_n$ Σ-termos e o um símbolo de operação n-ário; então $ot_1 \cdots t_n \in T_\Sigma\langle G\rangle$.

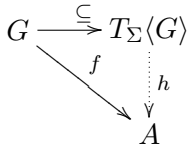
Os Σ-termos sobre G são portanto sempre *strings* de símbolos de operação e elementos de G. O conjunto G permite "gerar"mais termos e os seus elementos são designados por *geradores* (de $T_\Sigma\langle G\rangle$).

Exemplo. Sejam Σ a assinatura que apenas tem o símbolo de operação s (unário), e seja Σ' a assinatura que além deste tem também o símbolo 0 (constante). Tem-se $T_\Sigma = \emptyset$ e $T_\Sigma\langle\{0\}\rangle = T_{\Sigma'} = \{0, s0, ss0, \ldots\}$.

Dada uma assinatura Σ e um conjunto G, o conjunto $T_\Sigma\langle G\rangle$ tem, tal como T_Σ, uma estrutura de Σ-álgebra, onde para cada símbolo de operação n-ário e termos $t_1, \ldots t_n$ se tem $o_{T_\Sigma\langle G\rangle}(t_1, \ldots, t_n) \stackrel{\text{def}}{=} ot_1 \ldots t_n$. Esta álgebra designa-se por Σ-*álgebra livre gerada por* G.

Teorema. *Seja Σ uma assinatura, G um conjunto e A uma Σ-álgebra. Então, para cada função $f : G \to D_A$ existe um e um só homomorfismo $h : T_\Sigma\langle G\rangle \to A$ tal que $h(g) = f(g)$ para todos os geradores g.*

Dizemos habitualmente que h é o único homomorfismo que *estende* f, chamamos a h a *extensão homomórfica* de f, e representamos esta situação por meio do diagrama seguinte:



Prova. Semelhante à prova do Teorema A.6. Agora a base da definição indutiva tem duas partes: a dos símbolos de contante, como antes, e a dos geradores. ∎

Dada uma função $f : G \to D_A$ como acima, escreveremos habitualmente $t_{A,f}$ para o valor $h(t)$ dum termo t, generalizando a notação utilizada anteriormente.

Exemplo. Considere a assinatura Σ do exemplo anterior, e seja $f : \{0\} \to \mathbb{N}$ uma função, onde $\mathbb{N}$ é o conjunto dos números naturais equipado com a operação de *sucessor* ($n \mapsto n + 1$). Uma vez fixado o valor $f(0)$, a qualquer termo é atribuído um valor único pela extensão homomórfica de f. Por exemplo, se $f(0) = 100$ ter-se-á $f(s0) = 101$, $f(ss0) = 102$, etc.

A.8

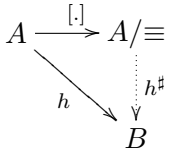
Seja Σ uma assinatura e G um conjunto. Uma *relação* em $T_\Sigma\langle G\rangle$ é um par $\langle t, u\rangle$, onde $t, u \in T_\Sigma\langle G\rangle$, usualmente escrito como uma equação "$t = u$".

Seja agora A uma Σ-álgebra e seja $f : G \to D_A$ uma função. Dizemos que a relação $t = u$ é *respeitada* por f se $t_{A,f} = u_{A,f}$. Do mesmo modo, se ρ é um conjunto de relações, dizemos que f respeita ρ se respeita todas as relações de ρ.

Um conjunto de relações ρ equivale a uma relação binária sobre $T_\Sigma\langle G\rangle$, o que nos permite falar da congruência gerada por ρ. A álgebra quociente $T_\Sigma\langle G\rangle/\equiv_\rho$ é designada por álgebra *apresentada por G e ρ*, e denotamo-la por $T_\Sigma\langle G \mid \rho\rangle$. A função que a cada gerador g faz corresponder a classe de equivalência $[g]$ é designada por *injecção de geradores*.

Lema. *Seja A uma Σ-álgebra e $\equiv$ uma congruência sobre A. Para qualquer homomorfismo $h : A \to B$ com a propriedade de que $h(x) = h(y)$ sempre que $x \equiv y$ existe um e um só homomorfismo $h^\sharp : A/\equiv \to B$ tal que para qualquer $[x] \in D_A/\equiv$ se tem $h^\sharp([x]) = h(x)$.*

Esta situação é representada pelo seguinte diagrama:



Prova. Seja B uma Σ-álgebra e $h : A \to B$ um homomorfismo. Seja $h^\sharp : D_A/\equiv \to D_B$ a função definida pela condição

$$h^\sharp([x]) \stackrel{\text{def}}{=} h(x) \ .$$

A função está bem definida porque se $[x] = [y]$, ou seja, $x \equiv y$, tem-se por hipótese $h(x) = h(y)$. Além disso, $h^\sharp$ é claramente a única função que satisfaz a condição, uma vez que a condição define a função em todo o seu domínio. Seja agora o um símbolo de operação n-ário e $x_1, \ldots, x_n \in D_A$. Tem-se

$$\begin{array}{rcll} h^\sharp(o_{A/\equiv}([x_1], \ldots, [x_n]) & = & h^\sharp([o_A(x_1, \ldots, x_n)]) & \text{(Pela Def. de } o_{A/\equiv}) \\ & = & h(o_A(x_1, \ldots, x_n) & \text{(Por hipótese)} \\ & = & o_B(h(x_1), \ldots, h(x_n)) & \text{(Porque } h \text{ é um homomorfismo)} \\ & = & o_B(h^\sharp([x_1], \ldots, [x_n])) & \text{(Por hipótese) .} \end{array}$$

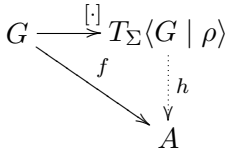
Portanto $h^\sharp$ é um homomorfismo $A \to B$, o que termina a demonstração. ∎

Lema. *Seja A uma Σ-álgebra e $\rho \subseteq D_A \times D_A$. Para qualquer homomorfismo $h : A \to B$ com a propriedade de que $h(x) = h(y)$ sempre que $x \rho y$ existe um e um só homomorfismo $h^\sharp : A/\equiv_\rho \to B$ tal que para qualquer $[x] \in D_A/\equiv_\rho$ se tem $h^\sharp([x]) = h(x)$.*

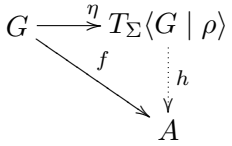
Prova. Seja $h : A \to B$ um homomorfismo tal que $x \rho y \Rightarrow h(x) = h(y)$. A relação $\equiv_h$ definida por $x \equiv_h y \iff h(x) = h(y)$ é uma congruência sobre A, com a propriedade $\rho \subseteq \equiv_h$, e portanto contém $\equiv_\rho$, pois esta é por definição a menor congruência que contém ρ. Por outras palavras, tem-se $x \equiv_\rho y \Rightarrow h(x) = h(y)$, e o resultado pretendido obtém-se por aplicação do lema anterior. ∎

Teorema. *Seja Σ uma assinatura, G um conjunto, ρ um conjunto de relações em $T_\Sigma\langle G\rangle$, e A uma Σ-álgebra. Então, para cada função $f : G \to D_A$ que respeita as relações existe um e um só homomorfismo $h : T_\Sigma\langle G \mid \rho\rangle \to A$ tal que $h([g]) = f(g)$ para todos os geradores g.*

Esta situação é representada pelo seguinte diagrama:



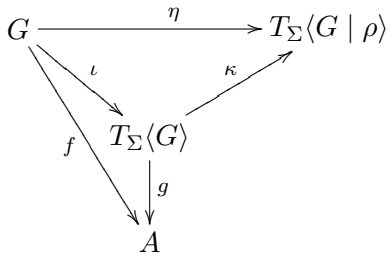
Prova. Nesta prova será útil usar nomes explícitos para as várias funções e homomorfismos que vão surgindo. Por exemplo, chamaremos η à função que a cada gerador $g \in G$ atribui a classe de equivalência $[g] \in T_\Sigma\langle G \mid \rho\rangle$, o que permite re-escrever o diagrama acima como se segue:



O enunciado do teorema pode ser reformulado equivalentemente dizendo que para cada função $f : G \to D_A$ que respeita as relações existe um e um só homomorfismo $h : T_\Sigma\langle G \mid \rho\rangle \to A$ tal que $h \circ \eta = f$.

Seja $\iota : G \to T_\Sigma\langle G\rangle$ a inclusão de G em $T_\Sigma\langle G\rangle$. Pelo Teorema A.7 existe um e um só homomorfismo $\kappa : T_\Sigma\langle G\rangle \to T_\Sigma\langle G \mid \rho\rangle$ tal que $\eta = \kappa \circ \iota$, e portanto κ é exactamente o homomorfismo que a cada termo $t \in T_\Sigma\langle G\rangle$ faz corresponder a classe de equivalência $[t]$. Seja agora A uma Σ-álgebra arbitrária e $f : G \to D_A$ uma função. Novamente pelo Teorema A.7, existe um homomorfismo único $g : T_\Sigma\langle G\rangle \to A$ tal que $f = g \circ \iota$. As várias funções e homomorfismos descritos até este momento estão representados no seguinte

diagrama:



Seja agora $h : T_\Sigma\langle G \mid \rho\rangle \to A$ um homomorfismo. Se $h \circ \kappa = g$ então $h \circ \eta = f$, pois

$$h \circ \eta = h \circ \kappa \circ \iota = g \circ \iota = f \ .$$

Por outro lado, $h \circ \eta = f$ é equivalente a $(h \circ \kappa) \circ \iota = f$. Mas já vimos que g é o único homomorfismo tal que $g \circ \iota = f$, e portanto tem de ter-se $h \circ \kappa = g$. Acabámos portanto de ver que dado um homomorfismo $h : T_\Sigma\langle G \mid \rho\rangle \to A$ as condições $h \circ \eta = f$ e $h \circ \kappa = g$ são equivalentes.

Finalmente, dizer que f respeita uma relação $t = u$ significa que $t_{A,f} = u_{A,f}$, ou seja, $g(t) = g(u)$. Pelo Lema anterior resulta então que existe um e um só homomorfismo $h : T_\Sigma\langle G \mid \rho\rangle \to A$ tal que $h \circ \kappa = g$, ou, equivalentemente, tal que $h \circ \eta = f$. ∎

A.9

Usualmente não trabalhamos com álgebras iniciais como acima. Por exemplo, a álgebra inicial correspondente a uma assinatura de semigrupo forma um grupóide e não um semigrupo, pois um semigrupo deve satisfazer também a propriedade associativa da multiplicação:

$$x \cdot (y \cdot z) = (x \cdot y) \cdot z \ .$$

Na equação acima x, y, z são *variáveis* que representam elementos arbitrários do semigrupo. Isto não significa que não exista um semigrupo inicial, i.e., um semigrupo a partir do qual existe um e um só homomorfismo de semigrupos para qualquer outro semigrupo. O nosso objectivo agora é tratar álgebras que obedecem a leis como a associatividade dos semigrupos, expressas por meio de variáveis.

Seja Σ uma assinatura e X um conjunto, cujos elementos designaremos por *variáveis*. [Assumiremos sempre que o conjunto de variáveis é disjunto do conjunto de símbolos de operação.] Uma Σ-*equação* sobre X, ou *lei algébrica* sobre X, é um par $\langle t, u\rangle$ de termos de $T_\Sigma\langle X\rangle$. Uma (Σ-)equação

em X é portanto o mesmo que uma relação em $T_\Sigma\langle X\rangle$; uma equação $\langle t, u\rangle$ é geralmente escrita na forma "$t = u$".

A.10

Seja agora A uma Σ-álgebra. Dizemos que a álgebra A *satisfaz* uma equação $t = u$ sobre X, e escrevemos $A \models t = u$, se para qualquer função $f : X \to D_A$ se tem $t_{A,f} = u_{A,f}$ (i.e., se qualquer função $f : X \to D_A$ respeita a equação—vista como uma relação sobre um conjunto de geradores X). Intuitivamente, $A \models t = u$ diz-nos que a equação é verdadeira independentemente dos valores de D_A que "atribuirmos"às variáveis.

Uma *especificação algébrica* sobre um conjunto X de variáveis é um par $\langle \Sigma, E\rangle$, onde Σ é uma assinatura e E é um conjunto (finito ou infinito) de Σ-equações sobre X. Uma Σ-álgebra A *satisfaz* a especificação $\langle \Sigma, E\rangle$, e escrevemos $A \models E$, se satisfaz todas as suas equações.

A.11

Seja $\langle \Sigma, E\rangle$ uma especificação algébrica e $\equiv$ uma congruência sobre T_Σ. Dizemos que uma congruência $\equiv$ sobre T_Σ *satisfaz* a especificação se $T_\Sigma/\equiv$ satisfaz a especificação.

Lema. *Seja $\{\equiv_i\}_{i\in I}$ uma família de congruências sobre $T_\Sigma\langle X\rangle$, para algum conjunto de indexação I. Então a congruência $\bigcap_{i\in I} \equiv_i$ satisfaz $\langle \Sigma, E\rangle$ se todas as congruências $\equiv_i$ o fizerem.*

Prova. Para cada $i \in I$ existe um e um só homomorfismo $u_i : T_\Sigma/\equiv \to T_\Sigma/\equiv_i$, onde escrevemos $\equiv$ em vez de $\bigcap_{i\in I} \equiv_i$; isto resulta de $\equiv \subseteq \equiv_i$ e do primeiro lema da Secção A.8 (tomando $A = T_\Sigma$) juntamente com o facto de que para cada $i \in I$ existe um e um só homomorfismo $j_i : T_\Sigma \to T_\Sigma/\equiv_i$; tem-se, para cada $i \in I$, $j_i = u_i \circ [.]$, onde $[.]$ é o único homomorfismo de T_Σ para $T_\Sigma/\equiv$. Sejam $[t]$ e $[u]$ elementos de $T_\Sigma/\equiv$. Tem-se $[t] = [u]$ sse $t \equiv u$ sse para qualquer $i \in I$ $t \equiv_i u$ sse para qualquer $i \in I$ $u_i([t]) = u_i([u])$, porque

$$t \equiv_i u \iff j_i(t) = j_i(u) \iff u_i([t]) = u_i([u]) \ .$$

Seja agora $t = u$ uma equação arbitrária de E, satisfeita por todas as congruências $\equiv_i$, $f : X \to T_\Sigma/\equiv$ uma função, e seja $h : T_\Sigma\langle X\rangle \to T_\Sigma/\equiv$ a extensão homomórfica de f. Tem-se $h(t) = h(u)$ sse, pelo que acabámos de ver, para qualquer $i \in I$ $u_i(h(t)) = u_i(h(u))$. Mas $u_i \circ h$ é um homomorfismo

de $T_\Sigma\langle X\rangle$ para $T_\Sigma/\equiv_i$ e portanto $u_i \circ h(t) = u_i \circ h(u)$ porque $T_\Sigma/\equiv_i$ satisfaz a equação. Como i é arbitrário resulta então que $h(t) = h(u)$. ∎

Seja $\langle \Sigma, E\rangle$ uma especificação algébrica. Do Lema resulta que existe a menor congruência que satisfaz a especificação, nomeadamente a intersecção de todas as congruências que a satisfazem. Designando esta congruência por $\equiv$, representaremos por $T_{\langle \Sigma,E\rangle}$ o quociente $T_\Sigma/\equiv$.

Teorema. *Seja $\langle \Sigma, E\rangle$ uma especificação algébrica e A uma álgebra que satisfaz a especificação. Então existe um e um só homomorfismo $h : T_{\langle \Sigma,E\rangle} \to A$.*

Prova. Seja $\iota_A : T_\Sigma \to A$ o único homomorfismo de T_Σ para A e seja $\equiv_A$ a congruência dada por

$$t \equiv_A u \iff \iota_A(t) = \iota_A(u) \ .$$

É simples ver que o (único) homomorfismo $i : T_\Sigma/\equiv_A \to A$ é injectivo e daí concluir que $T_\Sigma/\equiv_A$ satisfaz a especificação (pois dado um homomorfismo $h : T_\Sigma\langle X\rangle \to T_\Sigma/\equiv_A$ tem-se, para qualquer equação $t = u$, $i\circ h(t) = i\circ h(u)$—porque A satisfaz a equação—e pela injectividade de i resulta $h(t) = h(u)$). Isto significa que $\equiv_A$ satisfaz a especificação e portanto, pelo primeiro lema da Secção A.8, existe um homomorfismo único de $T_{\langle \Sigma,E\rangle}$ para $T_\Sigma/\equiv_A$; a unicidade resulta da unicidade dos homomorfismos a partir de T_Σ. Isto dá-nos um homomorfismo $T_{\langle \Sigma,E\rangle} \to A$, por composição com i, que é único devido à unicidade de i. ∎

Dada uma especificação $\langle \Sigma, E\rangle$ chamamos a $T_{\langle \Sigma,E\rangle}$ a $\langle \Sigma, E\rangle$-*álgebra inicial*.

Exemplo. Seja Σ a assinatura com símbolos "0"e "1"(constantes), "$-$"(unário), e "$+$"e "$\cdot$"(binários). Sejam x e y variáveis, e seja E o conjunto com as equações seguintes, onde escrevemos "xy"em vez de "$x \cdot y$", "$xy + z$"em vez

de "$(xy) + z$", etc.

$$\begin{aligned} x + (y + z) &= (x + y) + z \\ x + y &= y + x \\ x + 0 &= x \\ x + (-x) &= 0 \\ x(yz) &= (xy)z \\ x1 &= x \\ 1x &= x \\ x(y + z) &= xy + xz \\ (x + y)z &= xz + yz \end{aligned}$$

Uma $\langle \Sigma, E \rangle$-álgebra é um anel (com unidade). A álgebra inicial $T_{\langle \Sigma, E \rangle}$ é isomorfa ao anel $\mathbf{Z}$ dos números inteiros.

Nota. O modo de construir álgebras iniciais aqui exposto é diferente do que normalmente surge na literatura, e.g., em Johnstone [6] ou Baeten e Weijland [2].

Apêndice B

Exercícios diversos

Este apêndice contém uma lista de exercícios e problemas relacionados com os assuntos abordados na disciplina de Especificações Formais no ano lectivo de 1996/97, em parte ligados aos tópicos expostos nestas notas.

A primeira série (problemas) aborda tópicos de concorrência à-la-CCS (v. Milner [7]), e as duas seguintes incidem sobre teoria axiomática de conjuntos (v. Johnstone [6]).

B.1 Problemas

Problema 1—CCS interpretado em STAA

Seja $Act = \mathcal{L} \cup \{\tau\}$ como em CCS, e seja a assinatura $\mathsf{Proc}_{\mathrm{CCS}}$ o resultado de adicionar a Proc os seguintes símbolos de operação:

1. $|$, binário (*comunicação*);
2. $\backslash L$, unário, para cada $L \subseteq \mathcal{L}$ (*restrição*);
3. $[f]$, unário, para cada função de re-etiquetação f (*re-etiquetação*).

Tal como em CCS, usar-se-á a notação infixa para $|$ e pós-fixa para $\backslash L$ e $[f]$.

STAA, mantendo a interpretação habitual para as operações de Proc, é uma $\mathsf{Proc}_{\mathrm{CCS}}$-álgebra, onde os novos símbolos de operação são interpretados como se segue:

$$\begin{aligned} | &\mapsto \mathsf{com} \\ \backslash L &\mapsto \mathsf{res}_L \\ [f] &\mapsto \mathsf{rel}_f \end{aligned}$$

As operações com, res_L e rel_f são definidas do seguinte modo, para staas arbitrários $\mathsf{S} = \langle P, T, \imath\rangle$ e $\mathsf{T} = \langle Q, U, \jmath\rangle$:

$$\begin{array}{rcl} \mathsf{com}(\mathsf{S},\mathsf{T}) & \stackrel{\text{def}}{=} & \langle P \times Q, V, \langle \imath, \jmath\rangle\rangle \\ \mathsf{res}_L(\mathsf{S}) & \stackrel{\text{def}}{=} & \mathsf{staa}(\langle P, T \cap (P \times (Act \setminus (L \cup \overline{L})) \times P), \imath\rangle) \\ \mathsf{rel}_f(\mathsf{S}) & \stackrel{\text{def}}{=} & \langle P, \{\langle x, f(\alpha), y\rangle \mid \langle x, \alpha, y\rangle \in T\}, \imath\rangle \ , \end{array}$$

sendo V a menor relação de transição tal que

$$\begin{array}{rcl} (x \stackrel{\alpha}{\rightarrow}_T x') & \Rightarrow & \langle x, y\rangle \stackrel{\alpha}{\rightarrow} \langle x', y\rangle \\ (y \stackrel{\alpha}{\rightarrow}_U y') & \Rightarrow & \langle x, y\rangle \stackrel{\alpha}{\rightarrow} \langle x, y'\rangle \\ (x \stackrel{\ell}{\rightarrow}_T x' \ \text{ e } \ y \stackrel{\bar{\ell}}{\rightarrow}_U y') & \Rightarrow & \langle x, y\rangle \stackrel{\tau}{\rightarrow} \langle x', y'\rangle \end{array}$$

Mostre que a relação de bissimilaridade $\sim$ entre staas é uma congruência para as novas operações, e estude as propriedades algébricas de STAA/$\sim$. Em particular, verifique se as propriedades do CCS dão origem a equações satisfeitas por STAA/$\sim$; por exemplo, $P\backslash L \sim P$, com $\mathcal{L}(P) \cap (L \cup \bar{L}) = \emptyset$, é uma propriedade válida em CCS, e queremos saber se a equação $x\backslash L = x$ é ou não satisfeita em STAA/$\sim$.

Problema 2—Equivalência entre SOS e interpretações em STAA

A semântica operacional estrutural (SOS) para Proc atribui a cada termo de processo finito $P \in T_{\mathsf{Proc}}$ o staa

$$SOS(P) \stackrel{\text{def}}{=} \mathsf{staa}(\langle T_{\mathsf{Proc}}, Tr_{SOS}, P\rangle) \ ,$$

onde $Tr_{SOS} \subseteq T_{\mathsf{Proc}} \times Act \times T_{\mathsf{Proc}}$ é a relação de transição dada pela SOS.

Mostre que para qualquer termo de processo finito P se tem

$$SOS(P) \sim P_{\text{STAA}} \ .$$

Problema 3—Relação de transição sobre STAA

Dado um staa arbitrário $\mathsf{S} = \langle P, T, \imath\rangle$ e um estado $x \in P$, defina

$$\mathsf{S}(x) \stackrel{\text{def}}{=} \mathsf{staa}(\langle P, T, x\rangle) \ ,$$

e defina uma relação de transição sobre STAA tal que para quaisquer staas $\mathsf{S} = \langle P, T, \imath\rangle$ e $\mathsf{T} = \langle Q, U, \jmath\rangle$ se tenha $\mathsf{S} \stackrel{\alpha}{\rightarrow} \mathsf{T}$ sse existe $x \in P$ tal que

$\imath \xrightarrow{\alpha} x$ e $\mathsf{S}(x) \sim \mathsf{T}$. [Esta última condição pode ser abreviada, escrevendo simplesmente $x \sim \jmath$.]

Prove que dois staas S e T são bissimilares (i.e., existe uma bissimulação $R \subseteq P \times Q$ tal que $\imath R \jmath$) sse há uma bissimulação sobre STAA que os relaciona (i.e., existe uma bissimulação $S \subseteq \text{STAA} \times \text{STAA}$ tal que $\mathsf{S} S \mathsf{T}$).

Problema 4—Modelo F para Proc

Considere STAA equipado com a estrutura habitual de Proc-álgebra.

1. Prove que a equivalência de falhas $\sim_{\mathcal{F}}$ é uma congruência sobre STAA.

2. Estude as propriedades algébricas de $\mathbf{F} \stackrel{\text{def}}{=} \text{STAA}/\sim_{\mathcal{F}}$. Em particular, verifique quais dos axiomas de $\mathbf{T}$ são ainda satisfeitos em $\mathbf{F}$, e mostre que a equação $\alpha(\beta x + \beta y) = \alpha\beta x + \alpha\beta y$ é satisfeita em $\mathbf{F}$ mas não em STAA/$\sim$.

Problema 5—Composição paralela à esquerda

Recorde os axiomas para a composição paralela à esquerda:

1. $x \parallel y = x \between y + y \between x$
2. $\alpha x \between y = \alpha(x \parallel y)$
3. $(x + y) \between z = x \between z + y \between z$
4. $\mathbf{0} \between x = \mathbf{0}$
5. $x \between \mathbf{0} = x$
6. $(x \between y) \between z = x \between (y \parallel z)$

Defina uma interpretação apropriada para o símbolo $\between$ em STAA, de modo que $\sim$ seja uma congruência em STAA e os axiomas acima sejam satisfeitos em STAA/$\sim$. Enriqueça também a semântica operacional estrutural de Proc de modo a incluir a nova operação, e de tal modo que se verifiquem as seguintes propriedades:

1. $P \parallel Q \sim P \between Q + Q \between P$
2. $\alpha P \between Q \sim \alpha(P \parallel Q)$
3. $(P + Q) \between R \sim P \between R + Q \between R$

4. $\mathbf{0} \between P \sim \mathbf{0}$
5. $P \between \mathbf{0} \sim P$
6. $(P \between Q) \between R \sim P \between (Q \parallel R)$

Problema 6—Equações recursivas em STAA

Seja $\mathsf{S} = \langle P, T, \imath \rangle$ um staa e $x \in P$. Por definição de staa existe pelo menos um traço $t \in Act^*$ tal que $\imath \xrightarrow{t} x$ e definimos a *profundidade* de x como sendo

$$p(x) \stackrel{\text{def}}{=} \min\{\mathit{comprimento}(t) \mid \imath \xrightarrow{t} x\} \ .$$

Defina-se agora *corte de* S *de profundidade* n da seguinte forma:

$$c_n(\mathsf{S}) \stackrel{\text{def}}{=} \langle P_n, T_n, \imath \rangle \ ,$$

onde

$$\begin{aligned} P_n &\stackrel{\text{def}}{=} \{x \in P \mid p(x) \leq n\} \ , \\ T_n &\stackrel{\text{def}}{=} T \cap (P_n \times Act \times P_n) \ . \end{aligned}$$

1. Verifique que a definição é correcta, i.e., que dado um staa S, $c_n(\mathsf{S})$ é de facto um staa.
2. Prove que dois staas S e T são iguais sse $c_n(\mathsf{S}) = c_n(\mathsf{T})$ para qualquer $n \in \omega$.
3. Prove que a função $d : \mathrm{STAA}^2 \to \mathbb{R}$ dada por

   $$d(\mathsf{S}, \mathsf{T}) \stackrel{\text{def}}{=} \inf(\{2\} \cup \{\frac{1}{2^n} \mid c_n(\mathsf{S}) = c_n(\mathsf{T})\})$$

   é uma distância em STAA, i.e., que verifica as propriedades

   - $d(\mathsf{S}, \mathsf{T}) \geq 0$
   - $d(\mathsf{S}, \mathsf{T}) = 0$ sse $\mathsf{S} = \mathsf{T}$
   - $d(\mathsf{S}, \mathsf{T}) = d(\mathsf{T}, \mathsf{S})$
   - $d(\mathsf{S}, \mathsf{U}) \leq d(\mathsf{S}, \mathsf{T}) + d(\mathsf{T}, \mathsf{U})$ (desigualdade triangular)

   [Sugestão—há até uma propriedade mais forte que a desigualdade triangular: $d(\mathsf{S}, \mathsf{U}) \leq \max\{d(\mathsf{S}, \mathsf{T}), d(\mathsf{T}, \mathsf{U})\}$.]

4. Mostre que STAA com a distância d é um espaço métrico completo.

5. Seja x uma variável distinta dos símbolos de Proc e seja $P \in T_{\mathsf{Proc}}\langle\{x\}\rangle$. Mostre que se x é guardada em P (i.e., P é $\mathbf{0}$ ou da forma αQ ou de uma das formas $P_1 + P_2$ ou $P_1 \parallel P_2$, com x guardada em P_1 e P_2) então a equação $x = P$ tem uma solução única em STAA. [Sugestão: mostre que a função que interpreta P em STAA é uma contracção.]

Bibliografia auxiliar para espaços métricos completos (Cap. 2) e contracções (Apêndice 1): Simmons [9].

Problema 7—Correcção da SOS

Dado um staa arbitrário $\mathsf{S} = \langle P, T, \imath\rangle$ e um estado $x \in P$, defina

$$\mathsf{S}(x) \stackrel{\text{def}}{=} \mathsf{staa}(\langle P, T, x\rangle) \ ,$$

e defina uma relação de transição sobre STAA tal que para quaisquer staas $\mathsf{S} = \langle P, T, \imath\rangle$ e $\mathsf{T} = \langle Q, U, \jmath\rangle$ se tenha $\mathsf{S} \stackrel{\alpha}{\to} \mathsf{T}$ sse existe $x \in P$ tal que $\imath \stackrel{\alpha}{\to} x$ e $\mathsf{S}(x) \sim \mathsf{T}$. [Esta última condição pode ser abreviada, escrevendo simplesmente $x \sim \jmath$.]

Admitindo que dispõe dum conjunto de relações $R \subseteq T_{\mathsf{Proc}}\langle G\rangle \times T_{\mathsf{Proc}}\langle G\rangle$ tais que todos os termos de $T_{\mathsf{Proc}}\langle G\rangle$ têm interpretações únicas em STAA (i.e., para o qual existe uma e uma só atribuição de staas aos geradores que respeita as relações de R), mostre que a SOS é *correcta* face à relação de transição definida acima; isto é, mostre que, se $P, Q \in T_{\mathsf{Proc}}\langle G\rangle$ e se $P \stackrel{\alpha}{\to} Q$ pela SOS então $P_{\text{STAA}} \stackrel{\alpha}{\to} Q_{\text{STAA}}$ em STAA, onde P_{STAA} e Q_{STAA} são as interpretações (únicas) de P e Q em STAA. [Sugestão: utilize indução na profundidade das árvores de derivação para as transições.]

Nota—as regras da SOS para Proc, com os geradores de G e as relações de R, são as seguintes:

$$\textbf{Act} \quad \frac{}{\alpha P \stackrel{\alpha}{\to} P}$$

$$\textbf{Sum}_1 \quad \frac{P \stackrel{\alpha}{\to} P'}{P+Q \stackrel{\alpha}{\to} P'} \qquad \textbf{Sum}_2 \quad \frac{Q \stackrel{\alpha}{\to} Q'}{P+Q \stackrel{\alpha}{\to} Q'}$$

$$\textbf{Com}_1 \quad \frac{P \stackrel{\alpha}{\to} P'}{P \parallel Q \stackrel{\alpha}{\to} P' \parallel Q} \qquad \textbf{Com}_2 \quad \frac{Q \stackrel{\alpha}{\to} Q'}{P \parallel Q \stackrel{\alpha}{\to} P \parallel Q'}$$

$$\textbf{Con}_1 \quad \frac{\langle P,Q\rangle \in R \ , \ P \stackrel{\alpha}{\to} P'}{Q \stackrel{\alpha}{\to} P'} \qquad \textbf{Con}_2 \quad \frac{\langle P,Q\rangle \in R \ , \ Q \stackrel{\alpha}{\to} Q'}{P \stackrel{\alpha}{\to} Q'}$$

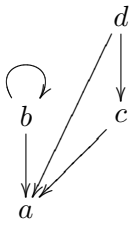
B.2 Teoria de conjuntos (ZF^-)

1. Suponha que a teoria de conjuntos apenas dispunha dos axiomas Ext e Emp_w.

   (a) Mostre que o (meta-)conjunto $\{a, b, c, d\}$, com a, b, c, d distintos entre si, e relação binária $\in$ definida por

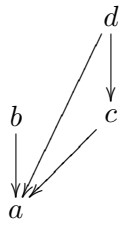
   $$a \in b\,,\ \ a \in c\,,\ \ b \in b\,,\ \ a \in d\,,\ \ c \in d\,,$$

   é um modelo para a teoria. Mostre que há um conjunto vazio (i.e., é satisfeito o axioma Emp_s).
   Nota: a relação $\in$ pode ser representada do seguinte modo:

   (b) A estrutura acima ainda seria um modelo na presença de
   i. Pair_w?
   ii. Sep?
   iii. Un_s?
   iv. Pow_w?
   v. Inf_w?

   (c) O mesmo conjunto, com a relação $\in$ definida por

   ainda é um modelo?

2. Prove que $(\forall x, y)(((x \subseteq y) \wedge (y \subseteq x)) \Rightarrow (x = y))$.

3. Prove os seguintes teoremas:

$$(\forall x, y, z)(\exists w)((x \in w) \wedge (y \in w) \wedge (z \in w)) \ ,$$
$$(\forall x, y, z)(\exists w)(\forall t)((t \in w) \iff ((t = x) \vee (t = y) \vee (t = z)) \ .$$

Justifique que um conjunto w nas condições da fórmula acima deve ser único. Qual a notação habitual para o conjunto w?

4. (a) Prove que não existe um conjunto cujos elementos sejam todos os conjuntos. [Sugestão: assuma que existe um conjunto v de todos os conjuntos e utilize o axioma da separação para obter uma contradição.]

   (b) Mostre que para qualquer conjunto a não existe nenhum conjunto b com a propriedade $(\forall x)((x \in b) \iff \neg(x \in a)$. [Isto é, mostre que não existem complementos de conjuntos.]

5. Sejam a e b conjuntos, com a não vazio, e f uma função tal que $f : a \to b$. Mostre que se f é injectiva, i.e. se

$$(\forall x, y, z)((((\langle x, z\rangle \in f) \wedge (\langle y, z\rangle \in f)) \Rightarrow (x = y)) \ ,$$

então existe uma função g tal que $g : b \to a$ e g é sobrejectiva, i.e.

$$(\forall x \in a)(\exists y \in b)(\langle y, x\rangle \in g) \ .$$

6. (a) Seja x um conjunto e f uma função tal que $f : x \to \mathcal{P}x$. Prove que f não é sobrejectiva. [Isto equivale a provar

$$(\forall f, x)((f : x \to \mathcal{P}x) \Rightarrow (\exists y)((y \in \mathcal{P}x) \wedge (\forall z \in x)(\neg(f(z) = y))) \ ,$$

   onde "$f(z) = y$"é uma abreviatura para $\langle z, y\rangle \in f$—este exercício mostra que o conjunto $\mathcal{P}x$ é estritamente "maior"que x.]

   (b) Seja x um conjunto e g uma função tal que $g : \mathcal{P}x \to x$. Prove que g não é injectiva.

7. Seja s a fórmula da teoria de conjuntos com variável livre x, definida por

$$s \equiv ((\emptyset \in x) \wedge (\forall t)((t \in x) \Rightarrow (t^+ \in x))) \ .$$

Diz-se que um conjunto x é um *conjunto de sucessores* se $(\emptyset \in x)$ e $(\forall t)((t \in x) \Rightarrow (t^+ \in x))$; o axioma Inf_w é precisamente a afirmação de que existe algum conjunto de sucessores, ou seja, é equivalente a

$(\exists x)s$. Prove que existe um conjunto de sucessores contido em todos os outros (versão forte de Inf), i.e. que

$$(\exists x)(s \wedge (\forall y)(s[y/x] \Rightarrow (x \subseteq y))) \ .$$

Justifique que um tal conjunto é único.

8. Seja p uma classe-função, com variáveis livres x e y. Mostre que para qualquer conjunto $a \subseteq \mathrm{dom}(p)$ existe uma função f tal que $\mathrm{dom}(f) = a$ e

$$(\forall x \in a)(\forall y)((\langle x, y\rangle \in f) \iff p) \ .$$

9. (Johnstone [6, Exercício 5.1]) Mostre que Pair e Sep podem ser deduzidos a partir de Emp, Pow e Rep.

10. Prove o seguinte teorema:

$$(\exists x)((\emptyset \in x) \wedge (\forall y)((y \in x) \Rightarrow (\{y\} \in x))) \ .$$

Sugestão: escreva uma fórmula p, com variáveis livres f e n, que exprima que f é uma função de domínio n^+ ($n \in \omega$), tal que $f(0) = \emptyset$ e $f(m^+) = \{f(m)\}$ para qualquer $m \in n$, e utilize-a para definir uma classe-função apropriada.

B.3 Teoria de conjuntos e processos

1. Seja $\mathsf{S} = \langle P, T, \imath\rangle$ um staa e α uma acção. Mostre que o conjunto de estados de $\mathsf{p}_\alpha(\mathsf{S})$ se pode construir em ZF^-.

2. Justifique que se existir um conjunto $\mathcal{E}$ de todos os estados então a classe de todos os sts é um conjunto (dado um conjunto de acções Act).

3. Seja $\mathsf{S} = \langle P, T, \imath\rangle$ um staa e α uma acção. Considere a seguinte definição alternativa de $\mathsf{p}\alpha$:

$$\mathsf{p}_\alpha(\mathsf{S}) = \langle P', T', \imath'\rangle \ ,$$

com $P' = P \cup \{P\}$, $T' = T \cup \{\langle P, \alpha, \imath\rangle\}$ e $\imath' = P$. Mostre que esta definição só é satisfatória na presença do axioma da fundação.

Bibliografia

[1] Peter Aczel. *Non-well-founded sets*. CSLI Lecture Notes 14. Center for the Study of Language and Information, 1988.

[2] J.C.M. Baeten and W.P. Weijland. *Process Algebra*. Cambridge University Press, 1990.

[3] Garret Birkhoff. *Lattice Theory*, volume 25 of *Colloquium Publications*. American Mathematical Society, 1967.

[4] B.A. Davey and H.A. Priestley. *Introduction to Lattices and Order*. Cambridge Mathematical Textbooks. Cambridge University Press, 1990.

[5] C.A.R. Hoare. *Communicating Sequential Processes*. Prentice Hall, 1985.

[6] Peter Johnstone. *Notes on Logic and Set Theory*. Cambridge Mathematical Textbooks. Cambridge University Press, 1987.

[7] Robin Milner. *Communication and Concurrency*. Prentice Hall, 1989.

[8] Robin Milner. *Communication and Mobile Systems: The π-Calculus*. Cambridge University Press, 1999.

[9] G.F. Simmons. *Topology and Modern Analysis*. McGraw-Hill, 1963.