Lição 3: Cálculo de Processos

Luís Soares Barbosa

Sumário

Nesta Lição vamos procurar responder à questão de saber em que circunstâncias dois processos, especificados na linguagem $\mathbb{P}$, representam o mesmo *comportamento. Cada resposta possível conduzir-nos-á a uma definição de* equivalência *entre processos. E, certamente, dispôr de uma noção de equivalência representa um salto fundamental no nosso arsenal de ferramentas de concepção e análise de sistemas reactivos e execução concorrente.*

De um ponto de vista técnico, a Lição introduz as teorias associadas às equivalências estrita e observacional. Discute-se ainda a existência de soluções para equações envolvendo estas equivalências e diversos exemplos de aplicação de raciocínio equacional à transformação de processos. A aplicação do cálculo será porém continuada na próxima Lição.

1 Motivação

Ao especificar um dado processo, já, provavelmente, nos interrogamos sobre o que é essencial e acessório nessa descrição. Por exemplo, será relevante a ordem das parcelas num contexto aditivo? Intuitivamente diremos que não; mas em que é que baseamos essa resposta? E o que se passará com a composição paralela? Será justificável igualar $E \mid \mathbf{0}$ com E? E com $\tau.E$?

Analogamente, quando manipulamos expressões aritméticas sabemos que podemos reordenar as parcelas de uma soma sem que isso afecte o resultado. Mas sabemos mais: sabemos a razão que nos permite proceder dessa forma — *o resultado final da adição é o mesmo em qualquer caso*. No cálculo de processos ainda não temos um conceito semelhante. Poderíamos arriscar dizer que as *transições envolvidas são as mesmas*, o que, sem dúvida, bastaria para justificar a comutatividade da escolha. Mas já não seria suficiente para justificar $a.\tau.E = a.E$.

Uma definição matemática de equivalência entre processos irá permitir-nos abordar estas questões. Questões que se tornam muito relevantes na prática. Suponhamos, por exemplo, que necessitamos substituir uma componente de um sistema, porque esta se esgotou ou se tornou demasiado cara. Será seguro substitui-la por uma componente diferente? Sem dúvida, desde que o comportamento do sistema permaneça essencialmente o mesmo. Uma noção precisa de equivalência permitirá provar que a substituição é ou não segura.

Mas que noção de equivalência será esta? Sabemos em que condições dois autómatos finitos são equivalentes: sempre que reconhecem a mesma linguagem (regular). Mas será essa uma noção de equivalência aceitável entre processos? Recordemos a discussão na Lição 2 sobre *sistemas de transição*. De facto, se pensarmos em processos como entidades dinâmicas que são observadas (*i.e.*,, interagem) enquanto se executam, faz sentido tomar todos os estados por que passam como estados terminais, uma vez que não podemos interromper a observação em algum momento particular. Deveremos, então, identificar os processos que realizam as mesmas sequências de acções? Ou bastará requerer a igualdade dos seus prefixos finitos? Ou do seu conteúdo observável? Mas será que os processos $a.(b.\,\mathbf{0} + c.\mathbf{0})$ e $a.b.\,\mathbf{0} + a.c.\mathbf{0}$, que tanto diferem na *quantidade*

Notas de apoio às aulas de *Interacção e Concorrência*, Lic. em Ciências da Computação, 2014-15.

de não-determinismo, são mesmo equivalentes? Ou a equivalência deve tomar em consideração não apenas as sequências de interacções mas também o seu padrão de ocorrência no tempo?

Por outro lado, teremos de reflectir sobre as propriedades que uma adequada noção de equivalência entre processos deve satisfazer. Dará origem a uma teoria equacional suficientemente expressiva? Será decidível? Será modular (*i.e.*,, congruente)?

2 Bissimilaridade e Equivalência Estrita

Definições

Processos são (casos particulares de) sistemas de transição. É, pois, natural que a noção de bissimulação estudada na Lição 2 seja tomada como a técnica base para a definição de relações de equivalência entre processos. Recordando essa definição, agora formulada no contexto do cálculo de processos que estamos a estudar, temos

Definição 1 *Uma relação binária S em $\mathbb{P}$ é uma* bissimulação (estrita) *sse, sempre que $(E, F) \in S$ e $a \in Act$,*

i) *se $E \xrightarrow{a} E'$ então $F \xrightarrow{a} F'$ e $(E', F') \in S$*

ii) *se $F \xrightarrow{a} F'$ então $E \xrightarrow{a} E'$ e $(E', F') \in S$*

Notemos que esta definição qualifica uma propriedade sobre relações em $\mathbb{P}$: em verdade existem diversas relações que a satisfazem. Vejamos alguns exemplos. Claramente a relação $\{(\mathbf{0}, \mathbf{0})\}$ satisfaz a definição dada; a prova é trivial na medida em que $\mathbf{0}$ não admite quaisquer transições. Podemos também verificar sem grande esforço que $\{(a.\mathbf{0}+a.\mathbf{0}, a.\mathbf{0}), (\mathbf{0}, \mathbf{0})\}$ é uma bissimulação. Note-se que esta relação não é simétrica nem reflexiva, o que mostra que nem todas as bissimulações são relações de equivalência. Por outro lado, para uma relação ser bissimulação não é suficiente conter apenas pares de processos idênticos. Por exemplo, $\{(a.\mathbf{0}, a.\mathbf{0})\}$ falha a definição porque o processo $a.\mathbf{0}$ admite uma transição por a para $\mathbf{0}$, mas o par $(\mathbf{0}, \mathbf{0})$ não pertence à relação.

Recordemos que a característica fundamental de uma bissimulação é o ser *fechada para a relação de transição*. I.e., as derivações de processos relacionados devem estar igualmente relacionadas entre si. Neste facto reside a razão pela qual a equivalência entre processos se pode basear neste tipo de relações. Se existir uma bissimulação que contenha o par formado pelos processos E e F, podemos concluir que estes se podem simular mutuamente ao longo de todas as suas derivações. A bissimulação em causa constitui, assim, uma *prova* de que os dois processos são indistinguíveis. Formalmente,

Definição 2 *Dois processos E e F são* estritamente equivalentes *(ou estritamente bissimilares), escrevendo-se $E \sim F$, sse existir uma bissimulação (estrita) S em $\mathbb{P}$ tal que $(E, F) \in S$.*

A definição dada é equivalente a dizer que

$$\sim \;=\; \bigcup \{S \subseteq \mathbb{P} \times \mathbb{P} |\; S \quad \text{é uma bissimulação (estrita)}\}$$

Nota 1 **[O Jogo da Bissimulação]**

A equivalência estrita pode ser estabelecida através de um jogo em que, dados dois processos E e F, dois jogadores discutem se as respectivas estruturas de transição são mutuamente correspondentes. Assim, o jogador I(nocêncio) tenta evidenciar essa correspondência, enquanto o jogador R(ezingão) procura mostrar o contrário. Uma partida deste jogo é iniciada por R escolhendo uma transição $E \xrightarrow{a} E'$ (ou $F \xrightarrow{a} F'$). Atento, I tenta encontrar uma transição com a mesma etiqueta no outro processo. Caso o consiga, R volta a jogar, I tenta de novo acompanha-lo, e assim sucessivamente.

R ganha a partida se I fôr incapaz de encontrar uma transição com etiqueta igual à da escolhida por R. Por seu lado, I é vencedor sempre que, tendo respondido a todas as jogadas de R, se chegou a uma situação em que os processos em causa já apareceram antes no decorrer dessa partida, ou em que R não pode jogar (porque os processos em causa já não admitem mais transições). Ganha, igualmente, se a partida puder continuar indefinidamente.

Um jogador vence o jogo se fôr capaz de vencer todas as partidas iniciadas com E e F. Nesse caso diz-se que o jogador em questão tem uma *estratégia vencedora*. Por exemplo, em

$$B \triangleq in.B \mid \overline{out}.B$$
$$B' \triangleq inB' + \overline{out}.B'$$

suponhamos que R escolheu a transição $B \xrightarrow{in} B$. I responde com $B' \xrightarrow{in} B'$. R escolhe, agora, $B' \xrightarrow{\overline{out}} B'$, respondendo I com $B \xrightarrow{\overline{out}} B$, ... Note-se que a partida nunca termina uma vez que I possui uma estratégia vencedora que consiste em copiar sempre a última jogada de R. Porque I possui uma estratégia vencedora dizemos que os processos B e B' são estritamente equivalentes (ou bissimilares). Notemos que uma estratégia vencedora para I significa, no caso geral, que

i) se R escolhe $E \xrightarrow{a} E'$ então I pode escolher $F \xrightarrow{a} F'$ e os processos E' e F' são ainda bissimilares.

ii) se R escolhe $F \xrightarrow{a} F'$ então I pode escolher $E \xrightarrow{a} E'$ e os processos E' e F' são ainda bissimilares.

Provas de Equivalência

A definição 2 fornece uma importante técnica de prova para demonstrar que dois processos são estritamente equivalentes: basta exibir uma bissimulação (estrita) que os contenha. Por exemplo, a equivalência $S \sim M$, para

$$T \triangleq i.\overline{k}.T$$
$$R \triangleq k.j.R$$
$$S \triangleq \mathsf{new}\ \{k\}\ (T \mid R)$$

e

$$M \triangleq i.\tau.N$$
$$N \triangleq j.i.\tau.N + i.j.\tau.N$$

decorre do facto $\langle S, M \rangle \in R$, onde R é a seguinte bissimulação

$$R = \{\langle S, M)\rangle, \langle \mathsf{new}\ \{k\}\ (\overline{k}.T \mid R), \tau.N \rangle, \langle \mathsf{new}\ \{k\}\ (T \mid j.R), N \rangle, \langle \mathsf{new}\ \{k\}\ (\overline{k}.T \mid j.R), j.\tau.N \rangle\}$$

O leitor poderá facilmente verificar que, pelo contrário, não é possivel construir uma bissimulação contendo o par formado pelos processos $a.(b.\ \mathbf{0} + c.\mathbf{0})$ e $a.b.\ \mathbf{0} + a.c.\mathbf{0}$.

Vamos, agora, considerar um outro exemplo, recordando a especificação de um semáforo feita anteriormente. Um semáforo é basicamente um dispositivo de controlo do acesso a um recurso critico de acordo com a expressão

$$Sem \ \triangleq \ get.put.Sem$$

Suponhamos, agora, que, num problema, necessitamos de um semáforo n-ário, *i.e.*,, um semáforo que admite qualquer sequência de get e put desde que a diferença entre o número dos primeiros e o número dos segundos não exceda n. Este tipo de semáforos é utilizado para controlar o acesso a recursos que admitem um número limitado de acessos simultâneos. Uma especificação directa deste processo será, por exemplo,

$$
\begin{aligned}
Sem_{n,0} &\triangleq get.Sem_{n,1} \\
Sem_{n,i} &\triangleq get.Sem_{n,i+1} + put.Sem_{n,i-1} \\
&\quad \text{(para } 0 < i < n\text{)} \\
Sem_{n,n} &\triangleq put.Sem_{n,n-1}
\end{aligned}
$$

Alternativamente, poderemos tentar realizar Sem_n pela composição paralela de n processos Sem, *i.e.*,,

$$Sem^n \;\triangleq\; Sem \mid Sem \mid ... \mid Sem$$

Intuitivamente, cada get sobre Sem^n corresponde a um get de uma das suas componentes, pelo que se torna possível realizar tantas destas acções quanto o número de componentes agregadas em paralelo. Uma vez esgotado o número de semáforos livres o sistema não dará mais permissões de acesso até que pelo menos um deles seja libertado.

Não é difícil mostrar que $Sem_{n,0} \sim Sem^n$. Por exemplo, para $n = 2$, chegamos rapidamente à seguinte bissimulação:

$$
\begin{aligned}
&\{\langle Sem_{2,0}, Sem \mid Sem\rangle, \langle Sem_{2,1}, Sem \mid put.Sem\rangle, \\
&\;\langle Sem_{2,1}, put.Sem \mid Sem\rangle \langle Sem_{2,2}, put.Sem \mid put.Sem\rangle\}
\end{aligned}
$$

Notemos que o segundo e o terceiro pares de S são *estruturalmente congruentes*. No entanto, para S ser de facto uma bissimulação não nos é possível prescindir de qualquer destes pares: se o fizessemos S deixaria de ser fechada para a relação de transição.

Este tipo de redundância, tantas vezes presente nas bissimulações, pode ser explorado por uma técnica expedita para reduzir a quantidade de trabalho envolvida na demonstração de uma equivalência. Suponhamos, para ilustrar a ideia, que retirarmos de S o par $(Sem_2(1), Sem \mid put.Sem)$. A relação resultante, embora não satisfaça a definição 1, constitui uma *bissimulação a menos de* $\equiv$, de acordo com

Definição 3 *Uma relação binária S em $\mathbb{P}$ é uma bissimulação (estrita) a menos de $\equiv$ sse, sempre que $(E, F) \in S$ e $a \in Act$,*

i) *se $E \xrightarrow{a} E'$ então $F \xrightarrow{a} F'$ e $(E', F') \in \equiv \cdot S \cdot \equiv$*

ii) *se $F \xrightarrow{a} F'$ então $E \xrightarrow{a} E'$ e $(E', F') \in \equiv \cdot S \cdot \equiv$*

onde $\cdot$ representa a composição de relações binárias. I.é, $(E, F) \in \equiv \cdot S \cdot \equiv$ sse existirem processos E' e F' tais que $E \equiv E'$, $(E', F') \in S$ e $F' \equiv F$. Não se requer, pois, que S seja fechada para as derivações, mas apenas que a cada derivação de um dos processos corresponda uma derivação do outro que conduza a um processo estruturalmente congruente com o primeiro.

O interesse da definição 3 reside na simplificação que acaba por introduzir nas provas. Voltando ao exemplo dos semáforos, necessitaríamos apenas de um conjunto de 3, e não 4, pares de processos. O leitor poderá verificar que, para um n arbitrário, a bissimulação que justifica $Sem_{n,0} \sim Sem^n$ contém 2^n pares enquanto uma bissimulação a menos de $\equiv$ necessita apenas de $n+1$ pares, o que representa, para n elevado, uma economia razoável. É claro que para podermos usar esta noção no estabelecimento de $\sim$ teremos, primeiro, de demonstrar que

Lema 1 *Se S é uma bissimulação (estrita) a menos de $\equiv$, então $S \subseteq \equiv$.*

Dispomos, assim, de uma poderosa técnica de prova: para estabelecer $E \sim F$, para F e E arbitrarios, é suficiente exibir uma bissimulação a menos de $\equiv$ que contenha o par (E, F). A demonstração do lema 1 baseia-se no seguinte resultado:

Lema 2 *Se S é uma bissimulação (estrita) a menos de $\equiv$, então $\equiv \cdot S \cdot \equiv$ constitui uma bissimulação (estrita).*

Prova. Seja S uma bissimulação (estrita) a menos de $\equiv$. Queremos provar que $\equiv S \equiv$ satisfaz as cláusulas i) e ii) de definição 1. Vamos, pois, assumir que $E \equiv S \equiv F$, para algum par de processos E e F. Suponhamos que $E \xrightarrow{a} E'$. Então, porque $E \equiv S \equiv F$, existem processos E_1 e F_1 tais que $E \equiv E_1$, $E_1 S F_1$ e $F_1 \equiv F$. De $E \xrightarrow{a} E'$ e $E \equiv E_1$ concluimos que existe um E_1' tal que $E_1 \xrightarrow{a} E_1'$ e $E' \equiv E_1'$. Agora, $E_1 \xrightarrow{a} E_1'$ e $E_1 S F_1$, juntamente com o facto de S ser uma bissimulação a menos de $\equiv$, permite-nos concluir que existe um F_1' que verifica $F_1 \xrightarrow{a} F_1'$ e $E_1' \equiv S \equiv F_1'$. Finalmente, se $F_1 \xrightarrow{a} F_1'$ e $F_1 \equiv F$, então $F \xrightarrow{a} F'$ e $F' \equiv F_1'$. Sumariando temos que, para algum E_1' e F_1' se verifica

$$F \xrightarrow{a} F' \qquad \text{e} \qquad E' \equiv E_1' \, S \, F_1' \equiv F'$$

A cláusula i) da definição de bissimulação segue daqui, dada a transitividade de $\equiv$. A prova da cláusula ii) usa o argumento simétrico.

$\square$

Podemos, agora, verificar o lema 1.

Prova. Se, como acabamos de mostrar, $\equiv \cdot S \cdot \equiv$ é uma bissimulação estrita, então $\equiv \cdot S \cdot \equiv \subseteq \sim$. Sabemos também que a relação de identidade em $\mathbb{P}$ está contida em $\sim$. Logo $\mathsf{id}_{\mathbb{P}} \cdot S \cdot \mathsf{id}_{\mathbb{P}} \subseteq \sim \cdot S \cdot \sim$, *i.e.*,, $S \subseteq \sim \cdot S \cdot \sim$ e, transitivamente, $S \subseteq \sim$, como queríamos.

$\square$

3 Teoria da Equivalência Estrita

Propriedades Básicas

Vamos, agora, investigar as leis satisfeitas pela relação $\sim$. Juntamente com os resultados das próximas secções, estas propriedades formam o núcleo do *cálculo* de processos que anunciamos no início deste curso e que R. Milner designou por CCS (acrónimo de *calculus of communicating systems*).

Ficamos, assim, equipados não apenas com uma metodologia para descrever problemas em níveis de abstracção distintos (e progressivamente mais detalhados), mas também com um cálculo equacional para discutir a sua equivalência (por exemplo, quando queremos mostrar a correcção da implementação de um processo P relativamente a uma especificação E). O cálculo permite-nos, nomeadamente, substituir, na prova de $E \sim P$, a construção explícita de uma bissimulação contendo E e P, por um argumento equacional.

Lema 3 *Sejam $E, F, G \in \mathbb{P}$. Então, $\langle \mathbb{P}, +, \mathbf{0} \rangle$ e $\langle \mathbb{P}, \mid, \mathbf{0} \rangle$ são monoides abelianos, sendo $+$ uma operação* *idempotente. Detalhadamente,*

$$\begin{aligned}
(E + F) + G &\sim E + (F + G)\\
E + F &\sim F + E\\
E + \mathbf{0} &\sim E\\
E + E &\sim E\\
(E \mid F) \mid G &\sim E \mid (F \mid G)\\
E \mid F &\sim F \mid E\\
E \mid \mathbf{0} &\sim E
\end{aligned}$$

Prova. As provas destas igualdades seguem, todas elas, um padrão similar (a que o leitor se deverá habituar realizando-as como exercício). Começa-se por conjecturar uma relação que contenha os processos cuja equivalência se quer demonstrar. De seguida verifica-se se a relação escolhida é, de facto, uma bissimulação por conferência das condições da definição 1 para todos os pares envolvidos. O único cuidado a ter é com a conjectura inicial. Ilustrativamente tentemos demonstrar a idempotência da escolha. Vamos começar por tomar a relação $S = \{(E + E, E) \mid E \in \mathbb{P}\}$. Ao tentar verificar as condições de (1) concluímos que, apesar de uma transição $E + E \xrightarrow{a} E$ ser correspondida por $E \xrightarrow{a} E$, o par (E, E) não pertence a S. Necessitamos, pois, de acrescentar a S todos os pares deste tipo, *i.e.*,, definimos S como $\{(E + E, E) \mid E \in \mathbb{P}\} \cup Id_{\mathbb{P}}$ que facilmente se verifica ser uma bissimulação. Note-se que aqui, como na maioria dos casos que encontraremos, a bissimulação que testemunha a equivalência procurada é uma relação infinita[1].

□

Lema 4 *Sejam $E, F \in \mathbb{P}$, $K \subseteq L$. Então,*

$$\begin{aligned}
\mathsf{new}\, K\ E &\sim E && \textit{se}\ \mathbb{L}(E) \cap (K \cup \overline{K}) = \emptyset\\
\mathsf{new}\, K'\ (\mathsf{new}\, K\ E) &\sim \mathsf{new}\, (K \cup K')\ E &&\\
\mathsf{new}\, K\ (E \mid F) &\sim \mathsf{new}\, K\ E \mid \mathsf{new}\, K\ F && \textit{se}\ \mathbb{L}(E) \cap \overline{\mathbb{L}(F)} \cap (K \cup \overline{K}) = \emptyset
\end{aligned}$$

Prova. A demonstração destas leis é feita, de novo, por construção de uma bissimulação (estrita) contendo os respectivos lados direito e esquerdo. Assim, por exemplo, para mostrar que

$$\mathsf{new}\, K\ E \ \sim\ E \qquad \text{se}\ \mathbb{L}(E) \cap (K \cup \overline{K}) = \emptyset$$

construimos, para um $K \subseteq Act$ arbitrário, a relação

$$S \ =\ \{(\mathsf{new}\, K\ E, E) \mid E \in \mathbb{P} \wedge \mathbb{L}(E) \cap (K \cup \overline{K}) = \emptyset\}$$

Para cada par $(\mathsf{new}\, K\ E, E) \in S$, se E transita por a para E', então $a, \overline{a} \notin K$, por definição de S. Logo as transições de qualquer elemento do par são correspondidas por transições idênticas do outro. Por outro lado, $\mathbb{L}(E') \subseteq \mathbb{L}(E)$, porque E' é uma derivação de E. Logo $\mathbb{L}(E') \cap (K \cup \overline{K}) = \emptyset$ o que permite concluir que $(\mathsf{new}\, K\ E', E') \in S$. Donde S é uma bissimulação (estrita), o que justifica a lei.

□

Um último comentário sobre as provas deste dois lemas torna-se necessário. Em alguns casos, nomeadamente nas leis que envolvem *apenas* operadores *estáticos*, a verificação pode resumir-se a concluir a igualdade entre os diagramas de sincronização dos respectivos lados direito e esquerdo. De facto, se não se verifica qualquer alteração nestes diagramas, que exibem as capacidades de interacção dos processos, podemos concluir a sua equivalência uma vez que estes operadores não são consumidos pelas transições. O argumento não se aplica, porém, quando estão envolvidos operadores dinâmicos. Por exemplo $a.\mathbf{0}$ e $a.a.\mathbf{0}$ têm o mesmo diagrama de sincronização mas não são, obviamente, equivalentes.

[1] O leitor atento verificará que ao provarmos que $\langle \mathbb{P}, +, \mathbf{0} \rangle$ e $\langle \mathbb{P}, \mid, \mathbf{0} \rangle$ são monoides abelianos estamos, de facto, a provar o resultado enunciado na Lição 3 que estabelece $\equiv$ como uma bissimulação.

Modularidade

Como se sabe, a bisimilaridade $\sim$ é uma relação de equivalência. Mas será que a substituição de um processo E por um processo F tal que $E \sim F$, num qualquer contexto, produz ainda resultados equivalentes? Notemos que se tal não fôr válido será quase impossível utilizar a noção de equivalência na prática: apenas se poderiam aplicar as leis anteriores ao nível mais externo das expressões e não às suas sub-expressões. A resposta, felizmente, é afirmativa: $\sim$ é compatível com os operadores em $\mathbb{P}$, ou seja, forma uma *congruência* em $\mathbb{P}$. De facto,

Lema 5 *Seja $E \sim F$. Então, para $P \in \mathbb{P}$, $K \subseteq L$ e f uma renomeação em L,*

$$a.E \sim a.F$$
$$E + P \sim F + P$$
$$E \mid P \sim F \mid P$$
$$\mathsf{new}\, K\ E \sim \mathsf{new}\, K\ F$$

Prova. Exercício 4

□

A prova segue o padrão das anteriores e é sugerida como exercício. Necessitamos ainda de dar mais um passo para estabelecer completamente o resultado que nos interessa. De facto, uma vez que admitimos a definição *recursiva* de processos torna-se necessário mostrar que este tipo de definições preserva a equivalência estrita. Comecemos, pois, por estender $\sim$ a processos com variáveis sobre $\mathbb{P}$. Suponhamos que E e F contém, no máximo, $\tilde{X}$ variáveis livres sobre $\mathbb{P}$. Então, $E \sim F$ sse, para todo a família de processos $\tilde{P}$, $\{\tilde{P}/\tilde{X}\}\, E \sim \{\tilde{P}/\tilde{X}\}\, F$. Neste contexto, tem-se

Lema 6

i) *Se $\tilde{P} \triangleq \tilde{E}$ então $\tilde{P} \sim \tilde{E}$, onde $\tilde{E}$ é uma família de expressões em $\mathbb{P}$ e $\tilde{P}$ uma família de identificadores de processos.*

ii) *Seja $\tilde{E} \sim \tilde{F}$, onde $\tilde{E}$ e $\tilde{F}$ são famílias de expressões recursivas em $\mathbb{P}$ sobre uma família $\tilde{X}$ de variáveis sobre $\mathbb{P}$. Definam-se $\tilde{A} \triangleq \{\tilde{A}/\tilde{X}\}\, \tilde{E}$ e $\tilde{B} \triangleq \{\tilde{B}/\tilde{X}\}\, \tilde{F}$. Então, $\tilde{A} \sim \tilde{B}$,* i.e.,, *$\sim$ é preservada pela definição recursiva.*

Prova. A primeira parte é trivial. Pela regra semântica ($ident$) sabemos que, para cada i, E_i e P_i têm exactamente as mesmas derivações, pelo que o resultado segue imediatamente. A prova da segunda parte, porém, requer indução sobre os diagramas de inferência (*cf.*[Mil89], páginas 99 e seguintes).

□

O Teorema da Expansão

A teoria da equivalência estrita que começamos a desenvolver inclui, ainda, um resultado fundamental sobre a relação entre processos definidos em termos da composição paralela e processos definidos por escolha não-determinista. Tal resultado é conhecido em CCS como o *teorema da* *expansão*, precisamente porque indica que a composição concorrente de processos pode ser expandida (*i.e.*,"desenrolada") numa escolha cujas parcelas são prefixadas pelas acções iniciais do processo composto.

Antes de enunciarmos a lei convirá recordar o exemplo atrás discutido onde concluímos, porventura com alguma surpresa, que $S \sim M$. Vejamos, agora, o resultado geral:

Lema 7 *Seja* $E \triangleq \mathsf{new}\, K\ (\{f_1\}\, E_1 \mid ... \mid \{f_n\}\, E_n)$, *com* $n \geq 1$. *Então,*

$$
\begin{aligned}
E \sim & \sum \{f_i(a).\mathsf{new}\, K\ (\{f_1\}\, E_1 \mid ... \mid \{f_i\}\, E'_i \mid ... \mid \{f_n\}\, E_n) \mid E_i \xrightarrow{a} E'_i \ \wedge\ f_i(a) \notin K \cup \overline{K}\} \\
& + \\
& \sum \{\tau.\mathsf{new}\, K\ (\{f_1\}\, E_1 \mid ... \mid \{f_i\}\, E'_i \mid ... \mid \{f_j\}\, E'_j \mid ... \mid \{f_n\}\, E_n) \mid E_i \xrightarrow{a} E'_i \ \wedge\ E_j \xrightarrow{b} E'_j \ \wedge\ f_i(a) = \overline{f_j(b)}\}
\end{aligned}
$$

Prova. Ver, por exemplo, [Mil89], páginas 96 e 97.

$\square$

Tomando substituições triviais, *i.e.*,, fazendo, na expressão anterior, $f_1 = f_2 = ... = f_n = \mathsf{id}$, chegamos a um caso particular do teorema com muito interesse. Assim,

$$
\begin{aligned}
E \sim & \sum \{a.\mathsf{new}\, K\ (E_1 \mid ... \mid E'_i \mid ... \mid E_n) \mid E_i \xrightarrow{a} E'_i \ \wedge\ a \notin K \cup \overline{K}\} \\
& + \\
& \sum \{\tau.\mathsf{new}\, K\ (E_1 \mid ... \mid E'_i \mid ... \mid E'_j \mid ... \mid E_n) \mid E_i \xrightarrow{a} E'_i \ \wedge\ E_j \xrightarrow{\overline{a}} E'_j\}
\end{aligned}
$$

Note-se, porém, que a formulação deste lema é notacionalmente pesada. Uma forma alternativa de pensar nele consiste em considerar o seguinte resultado (de resto, mais geral e, certamente, mais fácil de memorizar): *todo o processo é equivalente à soma das suas derivações imediatas*, *i.e.*,,

$$E \ \sim\ \sum \{a.E' \mid E \xrightarrow{a} E'\}$$

Outros casos particulares surgem no seguinte corolário:

Lema 8 *Para* $E, F \in \mathbb{P}$, $a \in Act$ *e* $K \subseteq L$, *verifica-se*

$$
\begin{aligned}
\mathsf{new}\, K\ (E + F) &\sim \mathsf{new}\, K\ E + \mathsf{new}\, K\ F \\
\mathsf{new}\, K\ (a.E) &\sim \begin{cases} \mathbf{0} & \textit{se } a \in (K \cup \overline{K}) \\ a.(\mathsf{new}\, K\ E) & \textit{caso contrário} \end{cases}
\end{aligned}
$$

Prova. Os resultados seguem por aplicação directa do lema 7, fazendo $n = 1$ e $f_1 = \mathsf{id}$.

$\square$

Vejamos, por fim, como o teorema se aplica ao exemplo anteriormente discutido, permitindo-nos deduzir por raciocínio equacional, $S \sim M$. Temos, pois,

$$
\begin{aligned}
S &\sim \mathsf{new}\, \{k\}\ (T \mid R) \\
&\sim i.\mathsf{new}\, \{k\}\ (\overline{k}.T \mid R) \\
&\sim i.\tau.\mathsf{new}\, \{k\}\ (T \mid j.R) \\
&\sim i.\tau.(i.\mathsf{new}\, \{k\}\ (\overline{k}.T \mid j.R) + j.\mathsf{new}\, \{k\}\ (T \mid R)) \\
&\sim i.\tau.(i.j.\mathsf{new}\, \{k\}\ (\overline{k}.T \mid R) + j.i.\mathsf{new}\, \{k\}\ (\overline{k}.T \mid R)) \\
&\sim i.\tau.(i.j.\tau.\mathsf{new}\, \{k\}\ (T \mid j.R) + j.i.\tau.\mathsf{new}\, \{k\}\ (T \mid j.R))
\end{aligned}
$$

Designando por N' a expressão $\mathsf{new}\,\{k\}\ (T \mid j.R)$ que, como vimos, expande em $N' {\sim} i.j.\tau.\mathsf{new}\,\{k\}\ (T \mid j.R) + j.i.\tau.\mathsf{new}\,\{k\}\ (T \mid j.R)$, podemos concluir as equivalências $N' \sim N$ e $S \sim i.\tau.N \sim M$ como queríamos[2].

Nota 2 **[O teorema da expansão e o paradigma da intercalação]**

O teorema da expansão, sustentando que a concorrência pode ser "explicada" em termos de não-determinismo, tem um significado muito profundo no cálculo de processos que temos vindo a introduzir. De facto, neste cálculo a concorrência é reduzida à *intercalação não-determinista* das sequências de acções em que o processo considerado se pode comprometer, tornando-se a sua caracterização semântica essencialmente uma teoria do não-determinismo. Esta abordagem à semântica dos processos, que é comum a CCS e outros cálculos (como CSP [Hoa85] ou ACP [BK85]), é justamente conhecida por *paradigma da intercalação* e é motivada pela hipótese de que a relação de causalidade entre acontecimentos não é observável. Ao sucesso deste paradigma não será alheio o facto de os sistemas de multiplexação de tarefas e utilizadores sobre uma mesma unidade de processamento ter sido uma das primeiras realizações da programação concorrente.

---

4 Equivalência Observacional

Vamos, agora, definir uma noção de equivalência insensível às acções não observáveis, *i.e.*,, que permite considerar equivalentes processos que diferem na quantidade de comportamento não observável que realizam. Veremos que este requisito tem de ser formulado cuidadosamente. Para isso comecemos por definir o que se entende por transições observáveis de um processo.

Transições Observáveis

A relação de transição básica que temos usado para caracterizar a semântica operacional dos processos é a família (*Act*-indexada) de relações $\xrightarrow{a}\ \subseteq\ \mathbb{P} \times \mathbb{P}$. Similarmente, podemos definir

$$\overset{a}{\Longrightarrow}\ \subseteq\ \mathbb{P} \times \mathbb{P}$$

como uma família de relações ditas de *transição observável*, em que as etiquetas são elementos de $L \cup \{\epsilon\}$. Uma $\overset{\epsilon}{\Longrightarrow}$-transição corresponde a zero ou mais transições elementares não observáveis (*i.e.*,, etiquetadas por τ). As seguintes regras de inferência definem a relação que nos interessa:

$$\frac{}{E \overset{\epsilon}{\Longrightarrow} E}\ (O_1)$$

$$\frac{E \xrightarrow{\tau} E' \quad E' \overset{\epsilon}{\Longrightarrow} F}{E \overset{\epsilon}{\Longrightarrow} F}\ (O_2)$$

$$\frac{E \overset{\epsilon}{\Longrightarrow} E' \quad E' \xrightarrow{a} F' \quad F' \overset{\epsilon}{\Longrightarrow} F}{E \overset{a}{\Longrightarrow} F}\ (O_3) \quad \text{para } a \in L$$

A importância de uma noção de equivalência em $\mathbb{P}$ baseada em transições observáveis pode ser

[2]Estas últimas substituições, parecendo intuitivas, são de facto justificadas por um resultado (o lema 17) sobre a existência de soluções únicas para equações recursivas que introduziremos mais adiante.

ilustrada com o exemplo seguinte. Consideremos os processos

$$
\begin{aligned}
T_0 &\triangleq j.T_1 + i.T_2 \\
T_1 &\triangleq i.T_3 \\
T_2 &\triangleq j.T_3 \\
T_3 &\triangleq \tau.T_0
\end{aligned}
$$

e

$$A \triangleq i.j.A + j.i.A$$

A análise dos seus grafos de transição

$$
\begin{tikzcd}
 & T_0 \arrow[dl, "j"'] \arrow[dr, "i"] & \\
T_1 \arrow[dr, "i"'] & & T_2 \arrow[dl, "j"] \\
 & T_3 \arrow[uu, bend right=80, "\tau"'] &
\end{tikzcd}
$$

e

$$
\begin{tikzcd}
 & A \arrow[dl, "j"'] \arrow[dr, "i"] & \\
i.A \arrow[ur, bend left=60, "i"] & & j.A \arrow[ul, bend right=60, "j"']
\end{tikzcd}
$$

respectivamente, permite concluir de imediato que que $T_0 \nsim A$. De facto, a transição $T_3 \stackrel{\tau}{\longrightarrow} T_0$ não tem correspondente no processo A. No entanto, a transição que os distingue é etiquetada por uma acção não observável. Podemos, pois, tentar formular uma equivalência entre processos que seja "cega" para as transições não observáveis. Tal equivalência irá igualar, por exemplo, transições como $E \stackrel{a}{\longrightarrow} E'$ com sequências de transições como $E \stackrel{\tau}{\longrightarrow}\stackrel{\tau}{\longrightarrow} \ldots \stackrel{a}{\longrightarrow} E'$ ou $E \stackrel{a}{\longrightarrow} \ldots \stackrel{\tau}{\longrightarrow}\stackrel{\tau}{\longrightarrow} E'$ ou, ainda, $E \stackrel{\tau}{\longrightarrow}\stackrel{\tau}{\longrightarrow} \ldots \stackrel{a}{\longrightarrow} \ldots \stackrel{\tau}{\longrightarrow}\stackrel{\tau}{\longrightarrow} E'$. Todas estas sequências de transições são, como sabemos, abstraídas em $E \stackrel{a}{\Longrightarrow} E'$. Assim, o que temos de fazer é apenas refazer as definições de bissimulação e equivalência em termos da relação de transição observável.

Equivalência Observacional

Para definir a relação de equivalência que nos interessa vamos simplesmente tomar a nova noção de *transição observável* e actualizar sobre ela a (sempre a mesma!) definição de *bissimulação*. Asim,

Definição 4 *Uma relação binária S em $\mathbb{P}$ é uma* bissimulação *(não estrita, ou fraca, ou observacional) sse, sempre que $(E, F) \in S$ e $a \in L \cup \{\epsilon\}$,*

i) *se $E \stackrel{a}{\Longrightarrow} E'$ então $F \stackrel{a}{\Longrightarrow} F'$ e $(E', F') \in S$*

ii) *se $F \stackrel{a}{\Longrightarrow} F'$ então $E \stackrel{a}{\Longrightarrow} E'$ e $(E', F') \in S$*

Evidentemente, como em qualquer bissimulação, temos que $\mathbf{1}_{\mathbb{P}}$, $\emptyset$, a inversa de uma bissimulação, a composição e a união de bissimulações, são, ainda, bissimulações de acordo com a definição acima. Podemos, agora, definir, à semelhança do que fizemos para $\sim$,

Definição 5 *Dois processos E e F são* (observacionalmente) equivalentes *(ou* bissimilares*), escrevendo-se $E \approx F$, sse existir uma bissimulação (observacional) S em $\mathbb{P}$ tal que $(E, F) \in S$.*

À luz desta definição podemos concluir, no exemplo dado no último comentário ao texto, que $T_0 \approx A$. Basta, para isso, mostrar que a relação

$$S = \{\langle T_0, A\rangle, \langle T_1, i.A\rangle, \langle T_2, j.A\rangle, \langle T_3, A\rangle\}$$

é uma bissimulação (não estrita).

Propriedades

As propriedades de $\approx$ são similares às que encontramos para $\sim$. Em particular, a definição de $\approx$ equivale a

$$\approx \; = \bigcup\{S \subseteq \mathbb{P} \times \mathbb{P} |\; S \text{ é uma bissimulação não estrita}\}$$

o que permite concluir que $\approx$ é a maior bissimulação (não estrita). Constitui, além disso, uma relação de equivalência. Discutiremos mais adiante a preservação de $\approx$ pelos operadores da linguagem.

Uma propriedade fundamental desta relação, que não é obviamente partilhada por $\sim$, é traduzida na lei

Lema 9 *Para qualquer processo $E \in \mathbb{P}$,*

$$E \;\approx\; \tau.E$$

Prova. Para toda a acção a tal que $E \xrightarrow{a} E'$ temos $\tau.E \xrightarrow{\tau} E \xrightarrow{a} E'$. I.é, $\tau.E \xRightarrow{a} E'$. Por outro lado, a única acção $\tau.E \xrightarrow{\tau} E$ é correspondida pela acção nula $E \xRightarrow{\epsilon} E$ em E. Claramente $\mathbf{1}_{\mathbb{P}} \cup \{(E, \tau.E) |\; E \in \mathbb{P}\}$ é uma bissimulação (não estrita).

□

Notemos, também, que, como se esperaria,

Lema 10 *Para quaisquer E e $F \in \mathbb{P}$, se $E \sim F$ então $E \approx F$,* i.e.,*, $\sim \subseteq \approx$.*

Prova. Exercício 10.

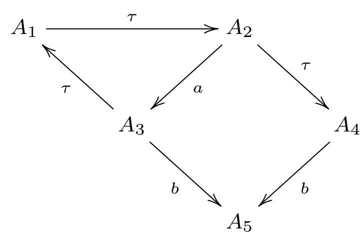
□

Nota 3 **[Construção de Bissimulações]**

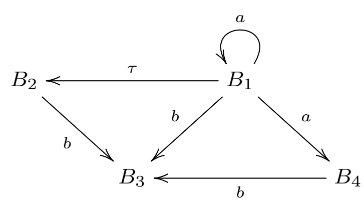
A construção de bissimulações constitui um procedimento necessário em muitas situações práticas. Justifica-se, por isso, um comentário um pouco extenso sobre um algoritmo de construção, aplicável a processos com um número finito de estados. Esta técnica baseia-se na análise dos grafos de transições e é, nomeadamente, usada no CWB-NC.

Consideremos, ilustrativamente os seguintes grafos de transições, que poderiam corresponder, por exemplo, aos processos

$$A_1 \triangleq \tau.(a.(b.\,\mathbf{0} + \tau.A_1) + \tau.b.\mathbf{0})$$
$$B_1 \triangleq a.B_1 + a.b.\,\mathbf{0} + \tau.b.\,\mathbf{0} + b.\mathbf{0}$$



e



O problema consiste em determinar quais os pares de estados bissimilares. A ideia desta técnica reside em procurar construir uma bissimulação iterativamente, a partir da relação universal sobre o conjunto de estados em causa, na qual todos os estados estão relacionados entre si. Desta forma temos a certeza de encontrar sempre a *maior* bissimulação contendo o par de processos em causa (caso exista, evidentemente!). Seguidamente, classificamos os estados de acordo com as suas derivadas imediatas para $\stackrel{a}{\Longrightarrow}$, para todo o $a \in L$. Estados equivalentes deverão ter as mesmas derivadas (observáveis).

Neste exemplo os estados A_1, A_2, A_3 e B_1 têm derivadas por $\stackrel{a}{\Longrightarrow}$ e por $\stackrel{b}{\Longrightarrow}$. Vamos agrupa-los num conjunto $C_1 = \{A_1, A_2, A_3, B_1\}$. Da mesma forma construimos os conjuntos $C_2 = \{A_4, B_2, B_4\}$ e $C_3 = \{A_5, B_3\}$, correspondendo aos estados que apenas têm derivadas por $\stackrel{b}{\Longrightarrow}$ (C_2) ou simplesmente não podem realizar qualquer transição observável (C_3).

Estes conjuntos definem uma relação binária nos estados: dois estados estão relacionados sse pertencerem ao mesmo conjunto. Vamos, agora, verificar se essa relação já será uma bissimulação. Como vimos, todo o estado em C_1 pode realizar $\stackrel{a}{\Longrightarrow}$ e evoluir para um estado também membro de C_1, facto que vamos registar escrevendo $C_1 \stackrel{a}{\Longrightarrow} C_1$. Da mesma forma concluimos que $C_1 \stackrel{a}{\Longrightarrow} C_2$, $C_1 \stackrel{\epsilon}{\Longrightarrow} C_2$, $C_1 \stackrel{b}{\Longrightarrow} C_3$ e $C_2 \stackrel{b}{\Longrightarrow} C_3$. Reparemos que cada uma destas asserções estabelece uma propriedade válida para todo o estado no conjunto à esquerda da seta; por outro lado, foram consideradas em cada caso todas as possíveis transições. Quer isto dizer que a relação *pertence ao mesmo conjunto* é, de facto, uma bissimulação, e, em particular $A_1 \approx B_1$. Note-se, por fim, que as transições por $\stackrel{\epsilon}{\Longrightarrow}$ dentro do mesmo conjunto (por exemplo, $A_3 \xrightarrow{\tau} A_1$) podem ser sempre trivialmente simuladas, a partir de qualquer estado no conjunto, pela inacção.

No caso de a relação encontrada não ser ainda uma bissimulação, a construção continua por divisão de um dos conjuntos. Para ilustrar isto mesmo vamos considerar uma nova transição etiquetada por a entre A_1 e A_5. A classificação dos estados permanece a mesma, mas agora temos $A_1 \stackrel{a}{\Longrightarrow} A_5$ e $A_3 \stackrel{a}{\Longrightarrow} A_5$, enquanto $A_5 \in C_3$. Temos, então, $C_1 \stackrel{a}{\Longrightarrow} C_3$ apenas para alguns estados em C_1, nomeadamente A_1 e A_3. Logo C_1 não pode ser uma classe de equivalência de uma bissimulação. Vamos, pois, gerar um novo conjunto C_4 composto pelos estados de C_1 que podem alcançar estados de C_3 por $\stackrel{a}{\Longrightarrow}$. I.é, $C_4 = \{A_1, A_3\}$.

Temos, agora, que verificar se o resultado forma uma bissimulação. De facto, temos $C_4 \stackrel{a}{\Longrightarrow} C_3$ e $C_4 \stackrel{\epsilon}{\Longrightarrow} C_1$. Não basta, contudo, verificar o novo conjunto de estados, mas todos aqueles com transições para C_4 ou C_1, uma vez que estes conjuntos foram alterados. O conjunto C_1 tem de ser revisto porque $A_2 \stackrel{a}{\Longrightarrow} A_3$, que pertence a C_4, enquanto B_1 não pode atingir estados em C_4. Torna-se, então, necessário dividir de novo o conjunto C_1, originando $C_1 = \{B_1\}$ e $C_5 = \{A_2\}$. As transições a partir de C_4 são, agora, $C_4 \stackrel{a}{\Longrightarrow} C_3$ e $C_4 \stackrel{a}{\Longrightarrow} C_4$ (através de C_5), $C_4 \stackrel{a}{\Longrightarrow} C_5$, $C_4 \stackrel{a}{\Longrightarrow} C_2$, $C_4 \stackrel{\epsilon}{\Longrightarrow} C_5$, $C_4 \stackrel{\epsilon}{\Longrightarrow} C_2$ e $C_4 \stackrel{b}{\Longrightarrow} C_3$. Os conjuntos C_5 e C_1 têm apenas um estado, não constituindo, portanto, problema. Os conjuntos C_2 e C_3 não possuem transições conducentes a estados dos conjuntos que foram divididos; estando correctos antes da divisão assim continuam. Concluimos, pois, que a nova relação é uma bissimulação.

Prosseguindo desta forma acabamos sempre por obter uma bissimulação, pois, no limite, o processo de divisão termina em conjuntos de estados singulares e a relação identidade é trivialmente uma bissimulação. O ponto importante é que obtemos sempre a *maior* bissimulação, na medida em que todas as tentativas maiores são eliminadas pelo processo de divisão. Assim, temos a certeza que estados no mesmo conjunto são observacionalmente equivalentes e, dualmente, estados em conjuntos distintos não o podem ser, porque, efectivamente, a maior bissimulação relaciona todos os estados equivalentes. No segundo exemplo considerado, temos, em particular $A_1 \not\approx B_1$.

Este método de construção de bissimulações oferece-nos, ainda, um bónus importante. De facto, se tomarmos cada conjunto C_i como nodo de um grafo cujos arcos são transições a partir de cada elemento de cada C_i, obtemos um grafo de transições equivalente mas com um número *minímo* de estados. No primeiro exemplo (sem a transição adicional por a) obtemos:

$$\begin{tikzcd}
C_1 \arrow[loop above, "a"] \arrow[r, "{\tau,a}"] \arrow[dr, "b"'] & C_2 \arrow[d, "b"] \\
 & C_3
\end{tikzcd}$$

Este grafo é mínimo em número de estados mas não no que concerne ao número de transições. De facto, as duas transições $C_1 \xrightarrow{a} C_2$ e $C_1 \xrightarrow{b} C_3$ são redundantes, *i.e.*,, simuláveis pelas restantes (como?). Se as removermos, chegamos a um grafo mínimo quer em termos de estados quer de transições:

$$\begin{tikzcd}
C_1 \arrow[loop above, "a"] \arrow[r, "\tau"] & C_2 \arrow[r, "b"] & C_3
\end{tikzcd}$$

O processo correspondente a C_1 pode ser escrito como

$$M \triangleq a.M + \tau.b.\mathbf{0}$$

Como o processo A_1 pertence a este conjunto, obtivemos, assim, uma representação minimal (e observacionalmente equivalente) a A_1.

Como referimos, os algoritmos de verificação de equivalência e minimização utilizados no CWB-NC operam exactamente deste modo. A mesma técnica pode ser usada para a construção de bissimulações estritas (e determinação de $\sim$) se se considerarem transições da forma $\xrightarrow{a}$ em vez de $\stackrel{a}{\Longrightarrow}$.

Modularidade

Tal como acima argumentamos, para ser útil no raciocínio sobre processos a equivalência $\approx$ deverá ser preservada pelos operadores do cálculo. Veremos de seguida que tal preservação falha para um desses operadores, o que nos motivará a introduzir uma pequena, mas importante, modificação no conceito de equivalência observacional. Comecemos por notar que, tal como acontece para $\sim$, temos que

Lema 11 *Seja $E \approx F$. Então, para $P \in \mathbb{P}$ e $K \subseteq L$*

$$a.E \approx a.F \tag{1}$$

$$E \mid P \approx F \mid P \tag{2}$$

$$\mathsf{new}\ K\ E \approx \mathsf{new}\ K\ F \tag{3}$$

Prova. Exercício 10.

□

No entanto, nas condições do lema anterior, já não se verifica $E + P \approx F + P$. I. é, a substituição de um processo E por um processo F, tal que $E \approx F$, num contexto aditivo, não leva a resultados $\approx$-equivalentes. Dito de outro modo, $\approx$ *não* é uma congruência.

Um exemplo ajudará a perceber o que se passa. Seja $P \triangleq i.\mathbf{0}$ e $Q \triangleq \tau.i.\mathbf{0}$. $P \approx Q$, uma vez que podemos igualar Q a $\tau.P$ e, como vimos, $E \approx \tau.E$, para todo o processo E. No entanto, não se verifica

$$j.\,\mathbf{0} + P \approx j.\,\mathbf{0} + Q$$

De facto, os grafos de transição

$$\begin{tikzcd} & j.\,\mathbf{0} + P \arrow[dl, "j"'] \arrow[dr, "i"] & \\ \mathbf{0} & & \mathbf{0} \end{tikzcd}$$

e

$$\begin{tikzcd} & j.\,\mathbf{0} + Q \arrow[dl, "j"'] \arrow[dr, "\tau"] & \\ \mathbf{0} & & i.\mathbf{0} \arrow[d, "i"'] \\ & & \mathbf{0} \end{tikzcd}$$

dos processos compostos, indicam que o segundo pode evoluir invisivelmente para um estado onde apenas pode realizar i. I. é, a transição $j.\,\mathbf{0} + Q \stackrel{\epsilon}{\Longrightarrow} i.\mathbf{0}$ corresponde à transição $j.\,\mathbf{0} + P \stackrel{\epsilon}{\Longrightarrow} j.\,\mathbf{0} + P$, mas claramente $i.\mathbf{0}$ e $j.\,\mathbf{0} + P$ não são bissimilares.

O exemplo anterior ilustra bem a razão pela qual $\approx$ não é uma congruência: o problema reside no τ inicial, em Q, que, no contexto aditivo, restringe o "menu" de opções possíveis.

O problema "desaparece" se, na definição sintáctica dos processos interactivos, se considerarem apenas somas guardadas (como em [Mil99]) em substituição do uso independente da *escolha* e do *prefixo*.

Lema 12 *Se $E \approx F$, então*

$$a.E + P \approx a.F + P$$
$$\mathsf{new}\ K\ E \approx \mathsf{new}\ K\ F$$
$$E \mid P \approx F \mid P$$
$$P \mid E \approx P \mid F$$

Prova. Exercício 10.

$\square$

Vamos, porém, adoptar a visão mais "clássica" que motiva a introdução de da seguinte definição,

Definição 6 *Dois processos E e F são observacionalmente congruentes (ou iguais), escrevendo-se $E = F$, sse*

i) $E \approx F$

ii) *se $E \xrightarrow{\tau} E'$ então $F \xrightarrow{\tau} F'' \stackrel{\epsilon}{\Longrightarrow} F'$ e $E' \approx F'$*

iii) *se* $F \xrightarrow{\tau} F'$ *então* $E \xrightarrow{\tau} E'' \stackrel{\epsilon}{\Longrightarrow} E'$ *e* $E' \approx F'$

A intuição sobre esta definição é de que toda a acção *inicial* não observável de E e F deve ser correspondida por, pelo menos, uma acção não observável do outro processo. É fácil mostrar que $=$ é uma congruência, inclusivamente para os contextos aditivos, e que, como seria de esperar, $E \neq \tau.E$. Tem-se, contudo, $\tau.E = \tau.\tau.E$ porque, após a transição inicial por τ, obtemos o par $(\mathbf{0}, \tau.\mathbf{0})$ de processos observacionalmente equivalentes. Repare-se que a definição de $=$ acima não é recursiva, nem é feita directamente em termos de bissimulações. De facto, as duas cláusulas da definição, com o requisito de que um τ deve ser correspondido por, pelo menos, outro τ, apenas são válidas ao primeiro nível das expressões; a partir daí usa-se a definição de equivalência observacional. Dito de outro modo, o requisito nas derivações é formulado em termos de $\approx$ e não de $=$.

Propriedades da Igualdade de Processos

A propriedade fundamental da relação $=$ reside, por um lado, na sua insensibilidade para toda a acção invisível (não inicial!), que herda da definição de $\approx$, e, por outro, na sua preservação pelos operadores em $\mathbb{P}$, em particular por $+$. De facto, um resultado interessante estabelece $=$ como uma restrição de $\approx$ aos pares de processos que a preservam em *todos* os contextos aditivos. Esta caracterização ilustra, igualmente, a proximidade entre as duas relações. A prova não é totalmente trivial e merece alguma atenção.

Lema 13 *Dados dois processos E e F cuja reunião das espécies é distinta do conjunto L de nomes de acções, $E = F$ sse, para todo o $G \in \mathbb{P}$,*

$$E + G \;\approx\; F + G$$

Prova. Comecemos por supôr que $E = F$. Basta-nos, pois, mostrar que

$$S \;\triangleq\; \{(E + G, F + G)|\; E, F, G \in \mathbb{P}\} \;\cup\; \approx$$

forma uma bissimulação. Temos de provar que S satisfaz as duas cláusulas da definição 4. Mostremos a primeira (a segunda segue o argumento simétrico). Suponhamos que $(U, V) \in S$ e que $U \xrightarrow{a} U'$. Se $U \approx V$ então $V \stackrel{a}{\Longrightarrow} V'$ e $U' \approx V'$ (porque $\approx$ é uma bissimulação não estrita), o que implica $(U', V') \in S$. Suponhamos, agora, e esta é a hipótese interessante, que U é da forma $E + G$, V é da forma $F + G$, com $E = F$. Há, neste caso, duas possibilidades de obter a transição $U \xrightarrow{a} U'$. A primeira consiste em $G \xrightarrow{a} G'$, mas, neste caso, tem-se igualmente $F + G \xrightarrow{a} G'$, *i.e.*,, $V \xrightarrow{a} G'$ e, como $G' \approx G'$, tem-se $(G', G') \in S$, como queríamos. A segunda possibilidade consiste em $E \xrightarrow{a} E'$. Então, de $E = F$, segue-se que $F \stackrel{a}{\Longrightarrow} F'$, com $E' \approx F'$. Logo, $F + G \stackrel{a}{\Longrightarrow} F'$ e $(E', F') \in S$.

Mostremos, agora, a implicação reversa por contra-recíproco (*i.e.*,, da negação da tese procuremos concluir a negação da hipótese). Suponhamos, pois, que $E \neq F$. Isto significa que existe pelo menos um a e um F' tais que $F \xrightarrow{a} F'$ mas, se $E \stackrel{a}{\Longrightarrow} E'$, então $E' \not\approx F'$. Vamos escolher $G \triangleq x.\mathbf{0}$, para x não pertencente nem à espécie de E nem de F. Notemos que é aqui que se torna necessária a hipótese do enunciado sobre as espécies de E e F. De facto, desconhece-se se o resultado é válido sem a hipótese. Note-se que, dada a admissibilidade em $\mathbb{P}$ de somas infinitas, os processos E e F poderiam, sem ela, usar todos os nomes de acções em L, mesmo para L infinito.

Prosseguindo, temos, pois, a transição $F + G \stackrel{a}{\Longrightarrow} F'$. Vamos, então, mostrar que, sempre que $E + G \stackrel{a}{\Longrightarrow} E'$, se tem $E' \not\approx F'$. Se $a = \tau$ e E' é da forma $E + G$, então o resultado segue porque E' pode realizar a acção x e F' não. Em qualquer outro caso, tem-se $E + G \stackrel{a}{\Longrightarrow} E'$, logo $E \stackrel{a}{\Longrightarrow} E'$, uma vez que a transição $G \stackrel{a}{\Longrightarrow} E'$ é impossível dado $a \neq x$. Então $E' \not\approx F'$.

□

Com base neste lema é muito simples provar a preservação de $=$ em contextos aditivos (que $\approx$ falha), *i.e.*,, $E + G = F + G$, desde que $E = F$.

Prova. A prova reduz-se, pelo lema anterior, a mostrar que, para todo o $G' \in \mathbb{P}$,

$$(E + G) + G' \;\approx\; (F + G) + G'$$

Sabemos que $E + (G + G') \approx F + (G + G')$, por aplicação do mesmo lema a $E = F$. Por outro lado, a escolha não-determinística é associativa para $\approx$, na medida em que já o era para a equivalência estrita $\sim$ que está contida em $\approx$. Logo a equação acima verifica-se.

□

Esta última prova remete-nos para a possibilidade de hierarquizar as várias noções de equivalência que introduzimos. Assim,

Lema 14 *Para quaisquer E e $F \in \mathbb{P}$, se $E \sim F$ então $E = F$. Similarmente, se $E = F$ então $E \approx F$,* i.e.,, $\sim \subseteq = \subseteq \approx$.

Prova. Exercício 21.

□

Note-se, porém, que da definição de $=$ decorre que, sendo E e F processos *estáveis*, *i.e.*,, incapazes de realizar τ como primeira acção, $E \approx F$ implica $E = F$. Reparemos, ainda, que $\approx$ pode ser caracterizada em termos da igualdade, de acordo com o resultado seguinte,

Lema 15 *Para quaisquer E e $F \in \mathbb{P}$, dizemos que $E \approx F$ sse $E = F$ ou $E = \tau.F$ ou $\tau.E = F$.*

Prova. Exercício 21.

□

Em resumo, a relação $=$, que vamos tomar como a noção de *igualdade* em $\mathbb{P}$, é uma congruência (inclusivamente com respeito à definição recursiva, caso em que se prova um resultado idêntico, mas formulado em termos de $=$, ao lema 6). Sem surpresas, a teoria equacional subjacente incorpora, de resto, todas as leis que estudamos para a equivalência estrita, incluindo o teorema da expansão. Além disso, tem-se especificamente

Lema 16 *Para quaisquer $E, F \in \mathbb{P}$ e $a \in L$ verificam-se as seguintes igualdades que permitem controlar a ocorrência de τ,*

$$a.\tau.E = a.E \tag{4}$$
$$E + \tau.E = \tau.E \tag{5}$$
$$a.(E + \tau.F) = a.(E + \tau.F) + a.F \tag{6}$$

Prova. A prova decorre trivialmente da definição 6.

□

Notemos, por fim, que, apesar da motivação e esforço feito na procura de uma congruência observacional, a simples equivalência ($\approx$) não deixa de ser útil. E isto porque não só $\approx$ é aplicável em muitas situações, mas sobretudo porque tem associada, tal como a equivalência estrita ($\sim$), uma poderosa técnica de prova — a construção de uma bissimulação contendo o par de processos em análise — que é automatizável para uma grande classe de processos (em particular, para processos de estado finito). Além disso os lemas anteriores mostram que, ao raciocinar sobre a igualdade de processos, podemos utilizar $\approx$ e introduzir $=$ apenas no "último passo" (porquê?).

5 Solução de Equações

É agora altura de discutir um resultado a que já fizemos referência e que concerne à *existência de soluçoes únicas* para equações recursivas nas estruturas $(\mathbb{P}, \sim)$ e $(\mathbb{P}, =)$. Em particular, o lema seguinte estabelece que tais soluções existem para sistemas de n equações recursivas sobre n variáveis em $\mathbb{P}$, desde que a ocorrência das variáveis satisfaça determinadas condições. A prova do lema não é fundamental para este curso (mas pode ser consultada em [Mil89], páginas 158 a 160); mais importante é não confundir este resultado com o lema 6, nem com o seu correspondente para a congruência observacional. Um (este!) é sobre *equações* recursivas, o outro concerne às *definições* recursivas e cai, consequentemente, no grupo de propriedades necessárias para o estabelecimento de uma congruência. Temos, pois,

Lema 17 *Equações recursivas em $\mathbb{P}$ da forma $\tilde{X} = \tilde{E}(\tilde{X})$ ou $\tilde{X} \sim \tilde{E}(\tilde{X})$ têm soluções únicas em $\mathbb{P}$ (a menos de $=$ ou $\sim$, respectivamente). Em rigor,*

i) *Seja $\tilde{E} = \{E_i | \ i \in I\}$ uma família de expressões com, no máximo, I variáveis livres ($\{X_i | \ i \in I\}$) tais que toda a variável que ocorre livremente numa expressão E_i é fracamente guardada. Então, se $\tilde{P} \sim \{\tilde{P}/\tilde{X}\}\tilde{E}$ e $\tilde{Q} \sim \{\tilde{Q}/\tilde{X}\}\tilde{E}$, tem-se que $\tilde{P} \sim \tilde{Q}$.*

ii) *Seja $\tilde{E} = \{E_i | \ i \in I\}$ uma família de expressões com, no máximo, I variáveis livres ($\{X_i | \ i \in I\}$) tais que toda a variável que ocorre livremente numa expressão E_i é guardada e sequencial. Então, se $\tilde{P} = \{\tilde{P}/\tilde{X}\}\tilde{E}$ e $\tilde{Q} = \{\tilde{Q}/\tilde{X}\}\tilde{E}$, tem-se que $\tilde{P} = \tilde{Q}$.*

Precisemos as condições sobre as variáveis. Uma variável é *fracamente guardada* numa expressão E sse ocorre numa sub-expressão de E da forma $a.E'$, para $a \in Act$. X é dita *guardada* sse a, na condição anterior, representar uma acção observável (*i.e.*,, $a \in L \cup \overline{L}$). Por exemplo, X é fracamente guardada em $\tau.X$ e $\tau.\,\mathbf{0} + a.X + b.a.X$, mas guardada apenas na segunda expressão. As condições em causa estabelecem, portanto, que o comportamento do processo é independente do processo que instancia a variável até à ocorrência de uma guarda.

No entanto, para equações recursivas em $=$ é ainda necessário impedir que a variável acabe por ficar guardada por um τ que resulte de uma sincronização. Tal seria o caso, por exemplo, de X em $X = \mathsf{new}\,\{a\}\ (\overline{a}.X \mid a.\mathbf{0})$. Impõe-se, assim, que toda a sub-expressão estrita de E (*i.e.*,, excluindo o próprio E) em que X ocorre seja da forma $a.E'$, para $a \in Act$, ou $\Sigma\tilde{E}$. Neste caso a ocorrência da variável diz-se *sequencial* na expressão E. Vejamos, de seguida, dois exemplos de aplicação deste resultado.

Exemplos

Relembremos o tratamento do problema da exclusão mútua através de um semáforo:

$$\begin{aligned}
Sem &\triangleq get.put.Sem \\
P_1 &\triangleq \overline{get}.c_1.\overline{put}.P_1 \\
P_2 &\triangleq \overline{get}.c_2.\overline{put}.P_2 \\
S &\triangleq \mathsf{new}\,\{get, put\}\ (Sem \mid P_1 \mid P_2)
\end{aligned}$$

e consideremos, agora, um processo $S' \triangleq \tau.c_1.S' + \tau.c_2.S'$ cuja equivalência com S queremos discutir.

Comecemos por reparar que S' é solução da equação $X = E(X)$, onde $E(X) = \tau.c_1.X + \tau.c_2.X$ é uma expressão sobre a variável X. De facto, pelo primeiro lema, $S' = \{S'/X\}E$. Tentemos verificar se S é igualmente uma solução da equação em causa. Assim, e começando por aplicar o teorema da expansão, obtemos:

$$\begin{aligned} S = \;& \tau.\mathsf{new}\, K\ (c_1.\overline{put}.P_1 \mid P_2 \mid put.Sem) + \tau.\mathsf{new}\, K\ (P_1 \mid c_2.\overline{put}.P_2 \mid put.Sem) \\ =\;& \tau.c_1.\mathsf{new}\, K\ (\overline{put}.P_1 \mid P_2 \mid put.Sem) + \tau.c_2.\mathsf{new}\, K\ (P_1 \mid \overline{put}.P_2 \mid put.Sem) \\ =\;& \tau.c_1.\tau.\mathsf{new}\, K\ (P_1 \mid P_2 \mid Sem) + \tau.c_2.\tau.\mathsf{new}\, K\ (P_1 \mid P_2 \mid Sem) \\ =\;& \tau.c_1.\tau.S + \tau.c_2.\tau.S \\ =\;& \tau.c_1.S + \tau.c_2.S \\ =\;& \{S/X\}E \end{aligned}$$

o que permite concluir $S = S'$, por aplicação do lema 17.

Vejamos, agora, um exemplo um pouco mais trabalhoso. Consideremos os seguintes processos:

$$\begin{aligned} B &\triangleq in.B_1 \\ B_1 &\triangleq in.B_2 + \overline{out}.B \\ B_2 &\triangleq \overline{out}.B_1 \end{aligned}$$

e

$$\begin{aligned} B' &\triangleq \mathsf{new}\, m\ (C_1 \mid C_2) \\ C_1 &\triangleq in.\overline{m}.C_1 \\ C_2 &\triangleq m.\overline{out}.C_2 \end{aligned}$$

Claramente o processo B é solução do seguinte sistema de 3 equações a 3 incógnitas (X, Y e Z):

$$\begin{aligned} X &= E(X,Y,Z) = in.Y \\ Y &= E_1(X,Y,Z) = in.Z + \overline{out}.X \\ Z &= E_3(X,Y,Z) = \overline{out}.Y \end{aligned}$$

através da substituição $\sigma = \{B/X, B_1/Y, B_2/Z\}$. I.é, $\tilde{B} = \sigma\tilde{E}$, dadas as famílias de processos $\tilde{B} = (B, B_1, B_2)$ e $\tilde{E} = (E, E_1, E_2)$.

Para estabelecer a equivalência $B = B'$, vamos, agora, procurar uma substituição σ' tal que $\tilde{B'} = \sigma'\tilde{E}$, para um $\tilde{B'} = (B', S_1, S_2)$. A questão que se coloca é saber como transformar a descrição do processo B' de forma a ser possível identificar as expressões para S_1 e S_2 em σ'. Para isso vamos recorrer, de novo, ao teorema da expansão. Assim,

$$\begin{aligned} B' = \;& \mathsf{new}\, m\ (C_1 \mid C_2) \\ =\;& in.\mathsf{new}\, m\ (\overline{m}.C_1 \mid C_2) \\ =\;& in.\tau.\mathsf{new}\, m\ (C_1 \mid \overline{out}.C_2) \\ =\;& in.\mathsf{new}\, m\ (C_1 \mid \overline{out}.C_2) \end{aligned}$$

Tentativamente, façamos $S_1 = \mathsf{new}\, m\ (C_1 \mid \overline{out}.C_2)$ e prossigamos a expansão:

$$\begin{aligned} S_1 = \;& \mathsf{new}\, m\ (C_1 \mid \overline{out}.C_2) \\ =\;& in.\mathsf{new}\, m\ (\overline{m}.C_1 \mid \overline{out}.C_2) + \overline{out}.\mathsf{new}\, m\ (C_1 \mid C_2) \\ =\;& in.\mathsf{new}\, m\ (\overline{m}.C_1 \mid \overline{out}.C_2) + \overline{out}.B' \end{aligned}$$

Façamos, por fim, $S_2 = \mathsf{new}\, m\ (\overline{m}.C_1 \mid \overline{out}.C_2)$. Então,

$$\begin{aligned} S_2 = &\ \mathsf{new}\, m\ (\overline{m}.C_1 \mid \overline{out}.C_2) \\ = &\ \overline{out}.\mathsf{new}\, m\ (\overline{m}.C_1 \mid C_2) \\ = &\ \overline{out}.\tau.\mathsf{new}\, m\ (C_1 \mid \overline{out}.C_2) \\ = &\ \overline{out}.\tau.S_1 \\ = &\ \overline{out}.S_1 \end{aligned}$$

A substituição pretendida fica, assim, completa, o que nos permite concluir a equivalência em causa entre B e B'.

Note-se que este problema pode ser reduzido ao da determinação de soluções de um sistema de 2 equações a 2 incógnitas, a saber,

$$\begin{aligned} X = &\ E(X,Y) = in.Y \\ Y = &\ E'(X,Y) = in.\overline{out}.Y + \overline{out}.in.Y \end{aligned}$$

Claramente, por substituição, obtemos

$$\begin{aligned} B = &\ in.B_1 \\ B_1 = &\ in.\overline{out}.B_1 + \overline{out}.in.B_1 \end{aligned}$$

Mostramos já que $B' = ... = in.S_1$. Prosseguindo o cálculo desta sub-expressão, obtemos

$$\begin{aligned} S_1 = &\ \mathsf{new}\, m\ (C_1 \mid \overline{out}.C_2) \\ = &\ in.\mathsf{new}\, m\ (\overline{m}.C_1 \mid \overline{out}.C_2) + \overline{out}.B' \\ = &\ in.\overline{out}.\mathsf{new}\, m\ (\overline{m}.C_1 \mid C_2) + \overline{out}.B' \\ = &\ in.\overline{out}.\tau.\mathsf{new}\, m\ (C_1 \mid \overline{out}.C_2) + \overline{out}.B' \\ = &\ in.\overline{out}.\tau.S_1 + \overline{out}.B' \\ = &\ in.\overline{out}.S_1 + \overline{out}.B' \\ = &\ in.\overline{out}.S_1 + \overline{out}.in.S_1 \end{aligned}$$

Logo, $B' = \{B'/X, S_1/Y\}E$ e $S_1 = \{B'/X, S_1/Y\}E'$. É imediato concluir que $B = B'$.

Referências

[BK85] J. Bergstra and J. Klop. Algebra for communicating processes with abstraction. *Theoretical Computer Science*, (37):77–121, 1985.

[Hoa85] C. A. R Hoare. *Communicating Sequential Processes*. Series in Computer Science. Prentice-Hall International, 1985.

[Mil89] R. Milner. *Communication and Concurrency*. Series in Computer Science. Prentice-Hall International, 1989.

[Mil99] R. Milner. *Communicating and Mobile Processes: the π-Calculus*. Cambridge University Press, 1999.

A Exercícios

Exercício 1

Indique quais das seguintes relações em $\mathbb{P}$ são bissimulações estritas.

$$S_1 = \{(\mathbf{0}, \mathbf{0})\}$$
$$S_2 = \emptyset$$
$$S_3 = \{(a.\mathbf{0}, a.\,\mathbf{0} + a.\mathbf{0}), (\mathbf{0}, \mathbf{0})\}$$
$$S_4 = \{(a.\mathbf{0}, a.\mathbf{0})\}$$
$$S_5 = \{(a.\mathbf{0} \mid b.\mathbf{0}, a.b.\,\mathbf{0} + b.a.\mathbf{0}), (\mathbf{0} \mid b.\mathbf{0}, b.\mathbf{0}), (a.\mathbf{0} \mid \mathbf{0}, a.\mathbf{0}), (\mathbf{0} \mid \mathbf{0}, \mathbf{0})\}$$

Exercício 2

Sejam $X \triangleq receive.send.X$ e $Y \triangleq send.receive.Y$. Mostre que $\{(send.X, Y), (X, receive.Y)\}$ é uma bissimulação estrita. Poderá concluir que $X \sim Y$?

Exercício 3

Indique quais dos seguintes processos são estritamente equivalentes a $a.b.\mathbf{0}$ (realize as provas e forneça os contra-exemplos necessários para justificar as suas conclusões.

1. $a.(b.\,\mathbf{0} + b.\mathbf{0})$
2. $a.b.\,\mathbf{0} + a.b.\mathbf{0}$
3. $a.\tau.b.\mathbf{0}$
4. $a.b.\,\mathbf{0} + a.\mathbf{0}$
5. $a.(b.\,\mathbf{0} + b.\mathbf{0}) + a.b.\mathbf{0}$
6. $a.b.\,\mathbf{0} + \mathbf{0}$
7. $a.\mathbf{0} \mid \mathbf{0}$
8. $\mathsf{new}\, \{c\}\; a.(b.\mathbf{0} \mid c.\mathbf{0})$
9. $\mathsf{new}\, \{c, d\}\; a.b.(c.\mathbf{0} \mid d.\mathbf{0})$

Exercício 4

Demonstre o lema 5.

Exercício 5

Suponha que alguém adicionou à linguagem de processos $\mathbb{P}$ dois outros operadores de composição paralela definidos pelas regras seguintes:

$$\frac{E \xrightarrow{a} E'}{E \otimes F \xrightarrow{a} E' \otimes F}\,(O_1) \qquad \frac{F \xrightarrow{a} F'}{E \otimes F \xrightarrow{a} E \otimes F'}\,(O_2)$$

$$\frac{E \xrightarrow{a} E' \quad \wedge \quad \overline{a} \notin \mathsf{fn}(F)}{E \parallel F \xrightarrow{a} E' \parallel F}\,(P_1) \qquad \frac{F \xrightarrow{a} F' \quad \wedge \quad \overline{a} \notin \mathsf{fn}(E)}{E \parallel F \xrightarrow{a} E \parallel F'}\,(P_2)$$

$$\frac{E \xrightarrow{a} E' \quad F \xrightarrow{\overline{a}} F'}{E \parallel F \xrightarrow{\tau} E' \parallel F'}\,(P_3)$$

1. Explique informalmente o propósito de $\otimes$ e $\|$, distinguindo-os da composição paralela que estudou.
2. A partir destas regras explique de que forma os diagramas de sincronização de $E \otimes F$ e $E \parallel F$ podem ser construidos a partir dos diagramas de E e de F. Compare esse processo com o que se passa com a construção do diagrama de sincronização de $E \mid F$.
3. Mostre ou refute a associatividade de $\|$ relativamente a $\sim$.

Exercício 6

Determine um processo P tal que $P \mid (a.b.\mathbf{0}) \sim a.b.a.\ \mathbf{0} + a.a.b.\mathbf{0}$, ou mostre que um tal P não pode existir.

Exercício 7

Considere o processo $A \triangleq a.(A \mid b.\mathbf{0})$.

1. Calcule o conjunto de derivações de A.
2. Prove que $A \mid A \sim A$.

Exercício 8

Para os seguintes pares de processos indique, justificando, os que podem ser relacionados por $\approx$. E por $=$?

1. $a.\tau.b.\mathbf{0}$ e $a.b.\mathbf{0}$
2. $a.(b.\ \mathbf{0} + \tau.c.\mathbf{0})$ e $a.(b.\ \mathbf{0} + c.\mathbf{0})$
3. $a.(b.\ \mathbf{0} + \tau.c.\mathbf{0})$ e $a.(b.\ \mathbf{0} + c.\mathbf{0}) + a.c.\mathbf{0}$
4. $a.\ \mathbf{0} + b.\ \mathbf{0} + \tau.b.\mathbf{0}$ e $a.\ \mathbf{0} + \tau.b.\mathbf{0}$
5. $a.\ \mathbf{0} + b.\ \mathbf{0} + \tau.b.\mathbf{0}$ e $a.\ \mathbf{0} + b.\mathbf{0}$
6. $a.(b.\ \mathbf{0} + (\tau.(c.\ \mathbf{0} + \tau.d.\mathbf{0})))$ e $a.(b.\ \mathbf{0} + (\tau.(c.\ \mathbf{0} + \tau.d.\mathbf{0}))) + a.(c.\ \mathbf{0} + \tau.d.\mathbf{0})$
7. $a.(b.\ \mathbf{0} + (\tau.(c.\ \mathbf{0} + \tau.d.\mathbf{0})))$ e $a.(b.\ \mathbf{0} + c.\ \mathbf{0} + d.\mathbf{0}) + a.(c.\ \mathbf{0} + d.\mathbf{0}) + a.d.\mathbf{0}$
8. $\tau.(a.b.\ \mathbf{0} + a.c.\mathbf{0})$ e $\tau.a.b.\ \mathbf{0} + \tau.a.c.\mathbf{0}$
9. $\tau.(a.\tau.b.\ \mathbf{0} + a.b.\tau.\mathbf{0})$ e $a.b.\mathbf{0}$
10. $\tau.(\tau.a.\ \mathbf{0} + \tau.b.\mathbf{0})$ e $\tau.a.\ \mathbf{0} + \tau.b.\mathbf{0}$
11. $A \triangleq a.\tau.A$ e $B \triangleq a.B$
12. $A \triangleq \tau.A + a.\mathbf{0}$ e $a.\mathbf{0}$
13. $A \triangleq \tau.A$ e $\mathbf{0}$

Exercício 9

Suponha que os processos R e T têm, entre outras, as transições seguintes: $R \xrightarrow{\tau} T$ e $T \xrightarrow{\tau} R$. Mostre que, nessa condições, se tem $R = T$.

Exercício 10

Demonstre os lemas 11, 10 e 12.

Exercício 11

Considere a definição seguinte de um *activador de n portas*:

$$AC_0 \triangleq \overline{s}.\mathbf{0}$$
$$AC_n \triangleq a.AC_{n-1}$$

1. Explique o comportamento deste processo.
2. Defina um activador de 4 portas usando apenas cópias do processo AC_2 e os operadores estáticos da linguagem.
3. Poderá o processo que definiu pode ser considerado observacionalmente equivalente a AC_4?

Exercício 12

Considere os factos seguintes acerca de uma relação binária S sobre $\mathbb{P}$. Para cada um deles, indique, justificando, se é ou não possível concluir que S constitui uma bissimulação observacional:

1. S é a relação identidade em $\mathbb{P}$.
2. S é um subconjunto da relação identidade em $\mathbb{P}$.
3. S é uma bissimulação estrita a menos de $\equiv$.
4. S é a relação vazia.
5. $S = \{(a.E, a.F) |\ E \approx F\}$.
6. $S = \{(a.E, a.F) |\ E \approx F\} \cup \approx$.

Exercício 13

Mostre que

1. $E + \tau.(E + F) \ = \ \tau.(E + F)$
2. $a.(E + \tau.\tau.E) \ = \ a.E$
3. $\tau.(G + a.(E + \tau.F)) \ = \ \tau.(G + a.(E + \tau.F)) + a.F$

Exercício 14

Considere a equação $X \ = \ a.\,\mathbf{0} + \tau.X$. Mostre que qualquer processo da forma $\tau.(\tau.P + a.\mathbf{0})$ é solução da equação. Considere, agora, o sistema de equações

$$\begin{aligned} X &= a.\,\mathbf{0} + \tau.Y \\ Y &= b.\,\mathbf{0} + \tau.X \end{aligned}$$

Como caracterizaria suas soluções ([Mil89], página 66) ?

Exercício 15

Considere a equação $X \ = \ \mathsf{new}\ \{a\}\ (a.X \mid \overline{a}.\mathbf{0})$, onde a variável X ocorre guardada, mas não sequencial, no lado direito. Mostre que o processo $\tau.P$ é solução desta equação, para qualquer P desde que $a, \overline{a} \notin \mathsf{fn}(P)$ ([Mil89], página 67).

Exercício 16

Para todo o processo E tal que $\mathsf{fn}(E) = \emptyset$, prove ou refute as seguintes afirmações:

1. $E \mid Q \approx Q$.
2. $E \mid Q = Q$.
3. $E \mid Q = \tau.Q$.

Exercício 17

Seja $E \triangleq a(x).\overline{a}\langle x \rangle.E$, *i.e.*,, E é um *buffer* de uma posição que utiliza a mesma porta para entrada e saída de valores. Assuma que esses valores são números inteiros.

1. Defina um processo sequencial (*i.e.*,, sem recurso à composição paralela) F tal que $F \approx E \mid E$.
2. Prove essa equivalência construindo uma bissimulação fraca relacionando os dois processos.
3. Será a sua proposta para F é minimal (no sentido de não conter acções não observáveis desnecessárias)?

Exercício 18

Recorde as definições de equivalência entre processos que estudou.

1. A partir de definição de $\approx$, mostre que se o processo A fôr definido como $\tau.A + B$ então $A \approx B$.
2. Dê um exemplo de processos X e Y não observacionalmente equivalentes, mas que verificam $X = \tau.X + Y$.
3. Seja $X \triangleq x.X + y.\mathbf{0}$ e $Y \triangleq \overline{x}.Y$. Mostre que $y.\ \mathbf{0}\ \approx \mathsf{new}\ \{x\}\ (X \mid Y)$.

Exercício 19

Seja P uma expressão na linguagem de processos contendo apenas uma variável livre X e considere as seguintes equações em X:

$$\begin{aligned} X &= \tau.(P + \mathsf{new}\ \{a\}\ (a.P \mid \overline{a}.\mathbf{0})) \\ X &= \tau.P \end{aligned}$$

Suponha ainda que alguém formulou a seguinte conjectura sobre estas equações:

Se $a \notin \cdots$, $\overline{a} \notin \cdots$ e $\cdots$ ocorrer guardada e sequencial em $\cdots$, as duas equações têm exactamente a mesma solução e esta é única (i.e.,, *todas as possiveis soluções são observacionalmente congruentes)*

1. Preencha as reticências na conjectura acima de modo a obter uma afirmação válida.
2. Prove a conjectura por raciocínio equacional.

Exercício 20

Considere os factos seguintes e, para cada um deles, indique, justificando, se é ou não possível concluir que $E = F$:

1. $E \approx F$ e E é um processo estável.
2. $E \approx F$ e nem E nem F são processos estáveis.
3. Existe um G tal que $E \mid G = F \mid G$.
4. $a.E = a.F$.
5. E e F satisfazem a mesma equação $X \sim E(X)$, onde as ocorrências de X em E são todas guardadas e sequenciais.

Exercício 21

Demonstre os lemas 14 e 15.

Exercício 22

Apesar de os sistemas concorrentes lidarem geralmente com processos perpétuos, em determinados casos é necessário considerar igualmente processos que realizam todas as suas tarefas e terminam com sucesso. Considere uma classe T de processos ditos *termináveis* que para indicar o termo da sua execução realizam uma acção observável especial †, após o que evoluem necessariamente para **0**.

Na classe T é possível definir um operador de composição *sequencial*, que representamos por $P \,;\, Q$, cujo significado intuitivo é: *após P terminar o processo composto comporta-se como Q*. Formalmente,

$$P \,;\, Q \quad \triangleq \quad \mathsf{new}\,(\{m/\dagger\}\, P \mid \overline{m}\cdot Q)\; m$$

sendo m um identificador de acção que não ocorre nem em P nem em Q.

1. Defina um processo $U \in T$ tal que $U \,;\, P \approx P$. Justifique a sua definição.
2. Mostre ou refute que, para $P, Q, R \in T$, se tem
$$(P + Q) \,;\, R \;\;\approx\;\; (P \,;\, R) \,+\, (Q \,;\, R)$$
3. Sendo a composição sequencial em T um caso particular da composição paralela, a lei anterior poderia parecer um caso particular da igualdade
$$(P + Q) \mid R \;\;\approx\;\; (P \mid R) \,+\, (Q \mid R)$$
No entanto esta igualdade é, em geral, falsa. Comprove esta afirmação fornecendo um contra-exemplo adequado.

Exercício 23

Considere a seguinte especificação, na linguagem de processos que estudou, da noção de *pipe* suportada no sistema UNIX:

$$U \rhd V \;\stackrel{\text{abv}}{=}\; \mathsf{new}\,\{c\}\;(\{c/out\}U \mid \{c/in\}V)$$

assumindo que, em ambos os processos, as acções $\overline{out}$ e in representam, respectivamente, as portas de saída e entrada.

1. Considere, agora, os seguintes processos, só parcialmente definidos:
$$\begin{aligned} U_1 &\triangleq \overline{out}.T \\ V_1 &\triangleq in.R \\ U_2 &\triangleq \overline{out}.\overline{out}.\overline{out}.T \\ V_2 &\triangleq in.in.in.R \end{aligned}$$
Prove, indicando sempre as leis que utilizou, ou refute as seguintes proposições:
   (a) $U_1 \rhd V_1 \;\;\sim\;\; T \rhd R$
   (b) $U_2 \rhd V_2 \;\;=\;\; U_1 \rhd V_1$
2. Mostre ou refute a associatividade do operador $\rhd$ relativamente à igualdade entre processos, *i.e.*,, para todo o $P, T, V \in \mathbb{P}$,
$$(U \rhd V) \rhd T \;\;=\;\; U \rhd (V \rhd T)$$
3. Mostre que $\mathbf{0} \rhd \mathbf{0} \;=\; \mathbf{0}$.

Exercício 24

Seja R uma bissimulação (observacional). Diga, justificando sucintamente, quais das afirmações seguintes pode concluir desse facto:

1. Se $(E, F) \in R$ então $E \approx F$.
2. Se $E \approx F$ então $(E, F) \in R$.
3. R é uma relação transitiva.

4. A composição de R consigo própria forma uma bissimulação estrita.

Exercício 25

Considere um operador $\circlearrowleft_n$ cuja semântica operacional é dada pela regra seguinte:

$$\frac{E \xrightarrow{a} E'}{\circlearrowleft_0 E \xrightarrow{a} E'} \qquad \frac{E \xrightarrow{a} E'}{\circlearrowleft_n E \xrightarrow{a} \circlearrowleft_{n-1} E'} \quad \text{para } n > 0$$

1. Indique sucintamente o seu propósito.
2. Discuta para que valores de m e n se poderá ter $\circlearrowleft_n (\circlearrowleft_m E) \sim \circlearrowleft_n E$.
3. Mostre que $E \sim F$ implica $\circlearrowleft_n E \sim \circlearrowleft_n F$.
4. Mostre, recorrendo a um contra exemplo, que a implicação da alínea anterior deixa de verificar-se se se substituir $\sim$ por $\approx$.
5. Como modificaria a semântica operacional deste novo operador para que a implicação referida se verificasse, *i.e.*,, $E \approx F \ \Rightarrow \ \circlearrowleft_n E \approx \circlearrowleft_n F$?

Exercício 26

Recorde o problema da Rede de Táxis discutido nas aulas teórico-práticas associadas à Lição 3 deste curso. Considere a seguinte especificação do seu comportamento tal como o "vê" o utente:

$$\begin{aligned}
\mathsf{Rede}(I) &\triangleq \mathsf{Vts}(I) \mid \mathsf{Atende} \\
\mathsf{Atende} &\triangleq \mathsf{tel}.\mathsf{Atende} \\
\mathsf{Vts}(I) &\triangleq |_{i \in I}\ \mathsf{V}(i) \\
\mathsf{V}(i) &\triangleq \mathsf{inicio}_i.\mathsf{fim}_i.\mathsf{V}(i)
\end{aligned}$$

onde o parâmetro I designa um conjunto finito de identificadores de viaturas, a acção tel modela o atendimento telefónico de um pedido de serviço e, por fim, as acções inicio_i e fim_i modelam, respectivamente, o início e o termo de um serviço prestado pela viatura i.

Suponha, agora, que numa cidade existem 3 redes de táxis, cada uma delas respeitando a especificação acima. O comportamento conjunto das 3 redes é dado por

$$\mathsf{RedeTotal} \triangleq \mathsf{Rede}(I_1) \mid \mathsf{Rede}(I_2) \mid \mathsf{Rede}(I_3)$$

onde I_1, I_2 e I_3 designa o conjunto de viaturas ligadas à rede 1, 2 e 3, respectivamente.

1. Verifique a validade da transição $\mathsf{RedeTotal} \xrightarrow{\mathsf{tel}} \mathsf{RedeTotal}$
2. Esboce o diagrama de sincronização do processo $\mathsf{RedeTotal}$
3. Discuta a validade da seguinte equação:
$$\mathsf{Atende} \mid \mathsf{Atende} = \mathsf{Atende}$$
4. Foi observado que para uma escolha apropriada de J se teria
$$\mathsf{RedeTotal} \ = \ \mathsf{Rede}(J)$$
Discuta a validade dessa proposta, indicando, caso exista, o valor J que torna veradeira a igualdade acima.
5. O resultado discutido na alínea 3. não é, como sabe, generalizável, *i.e.*,, a composição paralela de processos não é idempotente. Dê um exemplo de um processo P tal que $P \neq (P \mid P)$.

Exercício 27

---

onsidere o operador seguinte definido operacionalmente pela regra

$$\frac{E \xrightarrow{x} E'}{E \setminus\setminus_a \xrightarrow{x} E'} \text{ se } \quad x \neq a, x \neq \overline{a}$$

1. Indique sucinta mas claramente o seu propósito.
2. Prove, por construção de uma bissimulação, que se $P \sim Q$ então $P \setminus\setminus_a \sim Q \setminus\setminus_a$.
3. Defina dois processos E e F tais que $E \approx F$ mas $E \setminus\setminus_a \not\approx F \setminus\setminus_a$.
4. Recorrendo à definição de igualdade de processos, mostre ou refute que se $P = Q$ então $P \setminus\setminus_a = Q \setminus\setminus_a$.

Exercício 28

Considere um novo operador entre processos, dito *duplicador de acções*, e definido pela seguinte regra de inferência:

$$\frac{E \xrightarrow{a} E'}{\circlearrowleft (E) \xrightarrow{a} E}$$

Note que a derivada que surge na conclusão da regra é E e não E'. Por exemplo, $\circlearrowleft (a.\mathbf{0}) \xrightarrow{a} a.\mathbf{0}$. Prove ou refute que:

1. $E \sim F$ implica $\circlearrowleft (E) \sim \circlearrowleft (F)$.
2. $E \approx F$ implica $\circlearrowleft (E) \approx \circlearrowleft (F)$.
3. $\circlearrowleft (E + F) \sim \circlearrowleft (E) + \circlearrowleft (F)$.