



Linguaggi per il global computing

Esercizi B e D + Barbershop

Federico Perin - 2029215 - Ottobre 2021

Indice

1	Esercizi	2
1.1	Esercizio B	2
1.1.1	Sintassi	2
1.1.2	Dimostrazione per somme finite	2
1.1.2.1	Punto 1	2
1.1.2.1.1	Dimostrazione	2
1.1.2.2	Punto 2	4
1.1.2.2.1	Dimostrazione	4
1.1.2.2.2	Conclusione prima parte	6
1.1.3	Dimostrazione per somme finite	6

1 Esercizi

1.1 Esercizio B

Dimostrare che ogni processo CCS finito termina in un numero finito di passi.

1.1.1 Sintassi

$$P, Q ::= \alpha.P \mid (P \mid Q) \mid \sum_{i \in I} P_i \mid P \setminus L \mid P[f] \mid \mathbf{0}$$

Note: Nel CCS finito non sono previste le costanti \mathcal{K} . Il processo $\mathbf{0}$ ha un solo stato e non ha interazioni con altri processi.

La dimostrazione viene divisa in due casi: nel primo caso si hanno somme finite, cioè l'insieme I contenente le scelte della somma non deterministica, è finito. Nel secondo caso l'insieme I sarà infinito.

1.1.2 Dimostrazione per somme finite

La dimostrazione prevede i seguenti punti:

1. Ogni processo del CCS finito termina con un numero finito di passi;
2. Ogni processo del CCS finito ha un numero finito di stati.

1.1.2.1 Punto 1

Dato che non esistono costanti \mathcal{K} , non è possibile rigenerare passi eseguiti in precedenza, perciò dopo un certo numero finito di passi ogni processo P in CCS finito terminerà perché non avrà più passi da eseguire. Si deduce perciò che ogni processo ha un numero di passi limitato da un limite superiore è quindi tale numero è finito.

Per dimostrare quanto scritto si procede attraverso una dimostrazione induttiva sull'altezza di derivazione di un processo P , con ipotesi induttiva: $P \xrightarrow{\alpha} P' \Rightarrow \text{Size}_p(P') < \text{Size}_p(P)$, dove $\text{Size}_p(P')$ si intende il numero di passi del processo P dopo aver fatto l'azione α .

Definiamo Size_p :

$$\text{Size}_p(P) \begin{cases} P = \alpha.R, & 1 + \text{Size}_p R \\ P = \sum_{i \in I} P_i, & \max(\text{Size}_p P_i) \\ P = R \mid Q, & \text{Size}_p R + \text{Size}_p Q \\ P = R \setminus L, & \text{Size}_p R \\ P = R[f], & \text{Size}_p R \end{cases}$$

1.1.2.1.1 Dimostrazione

Caso Base:

$$\frac{}{\alpha.P \xrightarrow{\alpha} P \text{ ACT}} \quad \text{Size}_p(\alpha.P) = 1 + \text{Size}_p(P) > \text{Size}_p(P)$$

Si ha che l'azione α concatenata al processo P aggiunge un passo in più, perciò risulta essere corretto il limite superiore $\text{Size}_p(\alpha.P) = 1 + \text{Size}_p(P)$ passi.

Caso induttivo:

$$* \frac{P_j \xrightarrow{\alpha} P'}{\sum_{i \in I} P_i \xrightarrow{\alpha} P'} \text{ SUM } j \in I$$

Si osserva che nell'esecuzione di un passo α , per ipotesi induttiva vale che:
 $Size_p(P_j) > Size_p(P')$.

Perciò $Size_p(\sum_{i \in I} P_i) > Size_p(P_j) > Size_p(P')$ allora $max(Size_p P_i) > Size_p(P')$

Il limite superiore $Size_p(\sum_{i \in I} P_i) = max(Size_p P_i)$ passi, risulta essere corretto.

$$* \frac{P \xrightarrow{\alpha} P'}{P|Q \xrightarrow{\alpha} P'|Q} \text{ PAR-L}$$

Si osserva che nell'esecuzione di un passo α , per ipotesi induttiva vale che:
 $Size_p(P) > Size_p(P')$.

E quindi dato che $Size_p(P|Q) = Size_p(P) + Size_p(Q)$ mentre $Size_p(P'|Q) = Size_p(P') + Size_p(Q)$, per ipotesi induttiva $Size_p(P|Q) > Size_p(P'|Q)$

$$* \frac{Q \xrightarrow{\alpha} Q'}{P|Q \xrightarrow{\alpha} P|Q'} \text{ PAR-R}$$

Si osserva che nell'esecuzione di un passo α , per ipotesi induttiva vale che:
 $Size_p(Q) > Size_p(Q')$

E quindi dato che $Size_p(P|Q) = Size_p(P) + Size_p(Q)$ mentre
 $Size_p(P|Q') = Size_p(P) + Size_p(Q')$, per ipotesi induttiva
 $Size_p(P|Q) > Size_p(P|Q')$

$$* \frac{P \xrightarrow{\alpha} P' \quad Q \xrightarrow{\alpha} Q'}{P|Q \xrightarrow{\tau} P'|Q'} \text{ PAR-}\tau$$

Si osserva che nell'esecuzione di un passo di sincronizzazione, per ipotesi induttiva vale che: $Size_p(P) > Size_p(P')$ e $Size_p(Q) > Size_p(Q')$

E quindi dato che $Size_p(P|Q) = Size_p(P) + Size_p(Q)$ mentre
 $Size_p(P'|Q') = Size_p(P') + Size_p(Q')$, per ipotesi induttiva
 $Size_p(P|Q) > Size_p(P'|Q')$

Perciò si è dimostrato che il limite superiore $Size_p(P|Q) = Size_p(P) + Size_p(Q)$ passi, risulta essere corretto in tutti i 3 casi PAR-L, PAR-R e PAR- τ .

$$* \frac{P \xrightarrow{\alpha} P'}{P \setminus L \xrightarrow{\alpha} P' \setminus L} \text{ RES se } \alpha, \bar{\alpha} \in L$$

Si osserva che nell'esecuzione di un passo α , per ipotesi induttiva vale che:
 $Size_p(P) > Size_p(P')$

Dato che applicare una restrizione ad un processo non fa aumentare il numero massimo di passi di esecuzione, ma avere una possibile variazione delle possibili interazioni con altri processi in parallelo, vale che:
 $Size_p(P \setminus L) = Size_p(P)$ mentre $Size_p(P' \setminus L) = Size_p(P')$, per ipotesi induttiva $Size_p(P \setminus L) > Size_p(P' \setminus L)$

Perciò si è dimostrato che il limite superiore $Size_p(P \setminus L) = Size_p(P)$ passi, risulta essere corretto.

$$* \frac{P \xrightarrow{\alpha} P'}{P \setminus L \xrightarrow{\alpha} P' \setminus L} RES \text{ se } \alpha, \bar{\alpha} \in L$$

Nel caso in cui viene eseguito un passo $\alpha \in L$ perciò non è possibile eseguirlo, il processo P va in deadlock in un passo e quindi ha un numero finito di passi.

$$* \frac{P \xrightarrow{\alpha} P'}{P[f] \xrightarrow{f(\alpha)} P'[f]} REL$$

Si osserva che nell'esecuzione di un passo α , per ipotesi induttiva vale che:
 $Size_p(P) > Size_p(P')$

Dato che applicare un relabelling ad un processo non fa aumentare il numero massimo di passi di esecuzione, ma avere una possibile variazione delle possibili interazioni con altri processi in parallelo, vale che:
 $Size_p(P[f]) = Size_p(P)$ mentre $Size_p(P'[f]) = Size_p(P')$, per ipotesi induttiva $Size_p(P[f]) > Size_p(P'[f])$

Perciò si è dimostrato che il limite superiore $Size_p(P[f]) = Size_p(P)$ passi, risulta essere corretto.

1.1.2.2 Punto 2

Si vuole dimostrare che ogni processo CCS ha un numero finito di stati, cioè esiste un limite superiore del numero di stati di esecuzione per un processo P.

Definiamo $Size_s(P)$:

$$Size_s(P) \begin{cases} P = \mathbf{0}, & 1 \\ P = \alpha.R, & 1 + Size_s R \\ P = \sum_{i \in I} P_i, & \sum_{i \in I} Size_s P_i \\ P = R \mid Q, & Size_s R * Size_s Q \\ P = R \setminus L, & Size_s R \\ P = R[f], & Size_s R \end{cases}$$

Si dimostrerà di seguito che $Size_s(P)$ calcola il limite superiore del numero di stati dell'esecuzione del processo P.

1.1.2.2.1 Dimostrazione

Tramite dimostrazione per induzione sull'esecuzione di un processo P si dimostra che i limiti di $Size_s(P)$ sono corretti.

Caso Base:

0

Il processo **0** ha un solo stato, quello di partenza, per definizione quindi $Size_s(\mathbf{0}) = 1$

Caso induttivo:

In un passo si raggiunge uno sotto processo dal quale poi in n passi finiti si raggiungerà un stato terminante. Si utilizza la seguente ipotesi induttiva:

Un processo composto da sotto processi con un limite superiore ai stati di esecuzione ha anch'esso un limite superiore di stati di esecuzione.

Tale ipotesi scritta verrà utilizzata per dimostrare che il sotto processo raggiunto, da lui allo stato terminante, ci saranno un numero finiti di stati perché limitati da un limiti superiore. Si ricorda inoltre che il numero di stati raggiunti è finito se gli n passi sono finiti. Grazie alla dimostrazione del punto 1 si sa che n è finito.

* **ACT**

Con l'azione α concatenata al processo P viene aggiunto a P uno stato in più, infatti $\alpha.P \xrightarrow{\alpha} P$. P per ipotesi induttiva ha al più $Size_s(P)$ stati, quindi $\alpha.P$ avrà al più $Size_s(\alpha.P) = 1 + Size_s(P)$ stati.

* **SUM**

Si ha la seguente esecuzione: $\sum_{i \in I} P_i \xrightarrow{\alpha} P'$, dove qualunque P_i sia scelto per ipotesi induttiva ha un numero finito di stati d'esecuzione. Quindi tutti i processi che compongono la somma non deterministica hanno un numero finito di stati. Perciò $Size_s(\sum_{i \in I} P_i) = \sum_{i \in I} P_i$ stati, cioè il numero di stati del processo

P sarà al più $\sum_{i \in I} Size_s P_i$. Dato che il numero di processi P_i è finito anche la somma dei stati lo sarà.

È corretto il limite superiore presentato precedentemente perché potrebbero esserci stati non condivisi tra i vari processi P_i , quindi tutti i $Size_s(P_i)$.

* **PAR-L**

Si ha la seguente esecuzione: $P|Q \xrightarrow{\alpha} P'|Q$, dove $P|Q$ raggiunge lo stato $P'|Q$.

Per ipotesi induttiva $P'|Q$ ha un numero finito di stati d'esecuzione, cioè ha al più $Size_s(P'|Q) = Size_s(P') * Size_s(Q)$ stati, quindi per $P|Q$ al più il numero di stati sarà $Size_s(P|Q) = (Size_s(P') + 1) * Size_s(Q)$ stati. Dato che è possibile che non ci sia alcun stato condiviso durante l'esecuzione dei due processi, il limite superiore ad essi è uguale al totale delle combinazioni possibili tra gli stati dei due singoli processi.

* **PAR-R**

Si ha la seguente esecuzione: $P|Q \xrightarrow{\alpha} P|Q'$, dove $P|Q$ raggiunge lo stato $P|Q'$.

Per ipotesi induttiva $P|Q'$ ha un numero finito di stati d'esecuzione, cioè ha al più $Size_s(P|Q') = Size_s(P) * Size_s(Q')$ stati, quindi per $P|Q$ al più il numero di stati sarà $Size_s(P|Q) = Size_s(P) * (Size_s(Q') + 1)$ stati. Dato

che è possibile che non ci sia alcun stato condiviso durante l'esecuzione dei due processi, il limite superiore ad essi è uguale al totale delle combinazioni possibili tra gli stati dei due singoli processi.

* **PAR- τ**

Si ha la seguente esecuzione: $P|Q \xrightarrow{\tau} P'|Q'$, dove $P|Q$ raggiunge lo stato $P'|Q'$.

Per ipotesi induttiva $P'|Q'$ ha un numero finito di stati d'esecuzione, cioè ha al più $Size_s(P'|Q') = Size_s(P') * Size_s(Q')$ stati, quindi per $P|Q$ al più il numero di stati sarà $Size_s(P|Q) = Size_s(P') * Size_s(Q') + 1$ stati. Dato che è possibile che non ci sia alcun stato condiviso durante l'esecuzione dei due processi, il limite superiore ad essi è uguale al totale delle combinazioni possibili tra gli stati dei due singoli processi.

Dunque $Size_s(P|Q) = Size_s(P) * Size_s(Q)$ stati, risulta essere corretto.

* **RES**

Si ha la seguente esecuzione: $P \setminus L \xrightarrow{\alpha} P' \setminus L$.

Per ipotesi induttiva $P' \setminus L$ ha al più $Size_s(P' \setminus L) = Size_s(P')$ stati.

Dato che $Size_s(P' \setminus L)$ è finito, allora $P \setminus L$ avrà al più:

$Size_s(P \setminus L) = 1 + Size_s(P' \setminus L) = 1 + Size_s(P') = Size_s(P)$ stati, perché applicando una funzione di restrizione non si aumentano il numero massimo di stati d'esecuzione, vale quindi il limite superiore $Size_s(P \setminus L)$.

* **REL**

Si ha la seguente esecuzione: $P[f] \xrightarrow{f(\alpha)} P'[f]$.

Per ipotesi induttiva $P'[f]$ ha al più $Size_s(P'[f]) = Size_s(P')$ stati.

Dato che $Size_s(P'[f])$ è finito, allora $P[f]$ avrà al più:

$Size_s(P[f]) = 1 + Size_s(P'[f]) = 1 + Size_s(P') = Size_s(P)$ stati, perché applicando una funzione di relabelling non si aumentano il numero massimo di stati d'esecuzione ma si cambiano solo le possibili interazioni con altri processi in parallelo, vale quindi il limite superiore $Size_s(P[f])$.

1.1.2.2.2 Conclusione prima parte

Si è dimostrato nei punti precedenti che i processi definiti attraverso il CCS finito hanno sempre un limite superiore sia per il numero di passi e sia per il numero di stati d'esecuzione, di conseguenza il numero di passi e stati sono entrambi finiti.

1.1.3 Dimostrazione per somme finite

Ora per quanto riguarda la somma non deterministica $\sum_{i \in I} P_i$, l'insieme I sarà infinito.

La dimostrazione sarà simile a quella precedente per tutti i vari punti, tranne per il punto riguardante la somma non deterministica, infatti per dimostrare la finitezza dell'esecuzione si utilizzerà un'astrazione basata sulle generazioni della grammatica di CCS.

Si nota che i processi CCS che fanno parte della somma non deterministica, saranno infiniti e quindi avere un numero infinito di stati ma hanno tutti una lunghezza di derivazione finita.

Si dimostra perciò per induzione sulla lunghezza di derivazione che ogni processo CCS della somma non deterministica, termina in numero finito di passi.

Caso Base:

0

Il processo **0** termina in 0 passi per definizione.

Caso induttivo:

* **ACT, RES e REL**

* **PAR-L, PAR-R, PAR- τ**

* **SUM**