PROPRIETÀ DI LIMITAZIONE ALLA CRESCITA

Sia u il valore massimo delle variabili in ingresso del programma P e supponiamo che si sappia che $T_P(x_1, \dots, x_m) \leq f_n^{(k)}(u)$. Per calcolare un limite alla crescita dei valori delle variabili, consideriamo cosa accade ai valori delle variabili ad ogni passo del programma:

- Essere azzerati: assegnare 0 a una variabile non fa aumentare il suo valore.
- Avere assegnato il valore di un'altra variabile: può fare aumentare di molto il valore di una variabile, ma non aumenta il valore massimo delle variabili.
- Essere incrementati di una unità: l'aumento di un'unità, ripetuto, della variabile che ha il valore massimo comporta ovviamente l'aumento del valore massimo delle variabili.

Quindi se $T_P(x_1, \dots, x_m) \leq f_n^{(k)}(u)$, nella situazione più sfavorevole di un aumento di un'unità, ad ogni passo, della variabile che ha valore massimo u, il nuovo valore u' sarà:

$u' \leq u + T_P(x_1, \dots, x_m)$

$u + T_P(x_1, \dots, x_m) \leq u + f_n^{(k)}(u)$ perché $T_P(x_1, \dots, x_m) \leq f_n^{(k)}(u)$

$u + f_n^{(k)}(u) \leq f_n^{(k+1)}(u)$ per il lemma 8

TEOREMA DELLA LIMITAZIONE: Sia $P \in L_n$ allora esiste un k tale che $\mathbf{T_P(x_1, \dots, x_m) \leq f_n^{(k)}(u)}$ dove $\mathbf{u = \max(x_1, \dots, x_m)}$.

DIM (per induzione su n): Per n = 0 il programma P non ha cicli e quindi $T_P(x_1, \dots, x_m) = k$ (numero di istruzioni del programma), ma $k \leq f_0^{(k)}(u)$ per il lemma 2.

Assumiamo che il risultato sia vero per n-1 e dimostriamolo per n.

P, che appartiene a L_n, appartiene anche a L_{n-1} e si ha che:

- $T_P(x_1, \dots, x_m) \leq f_{n-1}^{(k)}(u)$ per l'ipotesi induttiva su n
- $f_{n-1}^{(k)}(u) \leq f_n^{(k)}(u)$ per il lemma 5

Sia P il programma seguente

LOOP V

Q con $Q \in L_{n-1}$

END

cioè vi è un solo ciclo esterno che porta ad n la profondità di nidificazione.

Distinguiamo due casi:

Se n = 1 allora $Q \in L_0$. In questo caso si avrà $T_Q = q$,
per cui $T_P(x_1, \dots, x_m) = qv$ (lunghezza di Q ciclata v volte) e si ha che:

$qv \leq qu$

$qu \leq (2^k)u$ per un opportuno k

$(2^k)u \leq f_1^{(k)}(u)$ per il lemma 9

Se n > 1, per l'ipotesi induttiva si ha $T_Q(x_1, \dots, x_m) \leq f_{n-1}^{(j)}(u)$ per qualche j. Un calcolo di P si ottiene facendo girare Q per v volte. <u>Dobbiamo però tener conto che ad ogni nuovo ciclo i valori delle variabili possono essere aumentati</u>. Dobbiamo perciò trovare valori che limitino la crescita possibile delle variabili dopo ogni esecuzione di Q.

La proprietà della limitazione alla crescita dei valori delle variabili in questo caso diviene:

$\mathbf{T_Q(x_1, \dots, x_m) \leq f_{n-1}^{(j)}(u) \quad \Rightarrow \quad u' \leq u + f_{n-1}^{(j)}(u) \leq f_{n-1}^{(j+1)}(u)}$

dove u' è il nuovo valore massimo delle variabili.

Facendo girare due volte il programma Q, il tempo di calcolo sarà dato dalla somma del tempo necessario a far girare Q con le variabili d'ingresso tali che $u = \max(x_1, \dots, x_m)$, che è dato da $f_{n-1}^{(j)}(u)$, e del tempo necessario a far girare Q quando le variabili di ingresso sono state modificate dalla precedente esecuzione di Q, che è limitato da $f_{n-1}^{(j)}(f_{n-1}^{(j+1)}(u))$.

Quindi il tempo di calcolo totale è limitato da:

$f_{n-1}^{(j)}(u) + f_{n-1}^{(j)}(f_{n-1}^{(j+1)}(u)) = f_{n-1}^{(j)}(u) + f_{n-1}^{(j+1)}(f_{n-1}^{(j)}(u))$

$f_{n-1}^{(j)}(u) + f_{n-1}^{(j+1)}(f_{n-1}^{(j)}(u)) \leq f_{n-1}^{(j+2)}(f_{n-1}^{(j)}(u))$ per il lemma 8

$f_{n-1}^{(j+2)}(f_{n-1}^{(j)}(u)) = f_{n-1}^{(2j+2)}(u)$

il valore massimo delle variabili d'ingresso sarà a questo punto limitato da $f_{n-1}^{(2j+3)}(u)$.

Quindi sia le variabili sia il tempo di calcolo sono limitate:

- dopo 1 giro di Q dà: $f_{n-1}^{(j+1)}(u)$
- dopo 2 giri di Q dà: $f_{n-1}^{(2j+3)}(u)$

dobbiamo quindi aggiungere all'esponente di f un numero pari a j+2 per volta.
Un buon limite per entrambi T e le variabili dopo u esecuzioni di Q è dato da $\mathbf{f_{n-1}^{(u(j+2))}(u)}$.
Allora abbiamo che:
$T_P(x_1, \dots, x_m) \leq T_{Q[u]}(x_1, \dots, x_m)$
$T_{Q[u]}(x_1, \dots, x_m) \leq f_{n-1}^{(u(j+2))}(u)$
$f_{n-1}^{(u(j+2))}(u) \leq f_{n-1}^{(u(j+2))}(f_n(u))$ per il lemma 3 e 4
$f_{n-1}^{(u(j+2))}(f_n(u)) = f_{n-1}^{(u(j+2))}(f_{n-1}^{(u)}(1))$ per definizione di f
$f_{n-1}^{(u(j+2))}(f_{n-1}^{(u)}(1)) = f_{n-1}^{(u(j+2)+u)}(1)$
$f_{n-1}^{(u(j+2)+u)}(1) = f_{n-1}^{(u(j+3))}(1)$
$f_{n-1}^{(u(j+3))}(1) \leq f_{n-1}^{(u(2\wedge(j+2))}(1)$ per il lemma 6
$f_{n-1}^{(u(2\wedge(j+2))}(1) \leq f_{n-1}^{(f1\wedge(j+2)(u))}(1)$ per il lemma 9
$f_{n-1}^{(f1\wedge(j+2)(u))}(1) \leq f_{n-1}^{(fn\wedge(j+2)(u))}(1)$ per il lemma 5 e 6
$f_{n-1}^{(fn\wedge(j+2)(u))}(1) = f_n(f_n^{(j+2)}(u)$ per definizione di f_n
$f_n(f_n^{(j+2)}(u) = f_n^{(j+3)}(u) = f_n^{(k)}(u)$ con k = j+3

Abbiamo così dimostrato il teorema per i programmi P della forma
LOOP
Q
END

Consideriamo ora il caso più generale in cui decomponiamo il programma in pezzi: ciascun pezzo sarà o un sottoprogramma che appartiene a L_{n-1}, oppure un sottoprogramma della forma considerata prima
LOOP
Q con $Q \in L_{n-1}$
END
Le variabili di uscita di ciascun sottoprogramma saranno le variabili di ingresso del sottoprogramma successivo. Quindi si avrà che:
$T_P(x_1, \dots, x_m) \leq f_n^{(k1)}(u) + f_n^{(k2)}(f_n^{(k1)}(u)) + f_n^{(k3)}(f_n^{(k2)}(f_n^{(k1)}(u))) + \dots + f_n^{(ks)}(\dots (f_n^{(k1)}(u)) \leq f_n^{(k)}(u)$ per un opportuno k, applicando il lemma 8 dopo aver osservato che il primo termine è l'argomento del secondo e così via.
Osserviamo cosa succede nel caso di una somma di tre soli termini:
$F = f_n^{(k1)}(u) + f_n^{(k2)}(f_n^{(k1)}(u)) + f_n^{(k3)}(f_n^{(k2)}(f_n^{(k1)}(u)))$
Ponendo $f_n^{(k1)}(u) = z$, la somma dei primi due termini diviene $z + f_n^{(k2)}(z)$ e si ha che
$z + f_n^{(k2)}(z) \leq f_n^{(k2+1)}(z) = f_n^{(k2+1)}(f_n^{(k1)}(u))$ per il lemma 8
Allora si ha che $F \leq f_n^{(k2+1)}(f_n^{(k1)}(u)) + f_n^{(k3)}(f_n^{(k2)}(f_n^{(k1)}(u)))$.
In questo modo il primo termine, cioè $f_n^{(k2+1)}(f_n^{(k1)}(u))$, non è argomento della funzione successiva $f_n^{(k3)}$.
Però osservando che
$f_n^{(k3)}(f_n^{(k2)}(f_n^{(k1)}(u))) \leq f_n^{(k3)}(f_n^{(k2+1)}(f_n^{(k1)}(u)))$ per il lemma 4, poiché
$f_n^{(k2)}(f_n^{(k1)}(u)) < f_n^{(k2+1)}(f_n^{(k1)}(u))$ per il lemma 6
allora si ha che
$F < f_n^{(k2+1)}(f_n^{(k1)}(u)) + f_n^{(k3)}(f_n^{(k2+1)}(f_n^{(k1)}(u)))$ e in questo modo il primo termine è argomento di $f_n^{(k3)}$.
Ora applicando il lemma 8 otterremo che
$F < f_n^{(k3+1)}(f_n^{(k2+1)}(f_n^{(k1)}(u))) = f_n^{[(k3+1)+(k2+1)+(k1)]}(u)$
Il meccanismo resta invariato per una somma di s termini per cui si ha che T_P è maggiorato da:
$f_n^{[(ks+1)+\dots+(k2+1)+(k1)]}(u) = f_n^{(k1+k2+\dots+ks+s-1)}(u)$

COROLLARIO: Se $g \in \mathcal{L}_n$ allora esiste una costante k tale che $\mathbf{g(x_1, \dots, x_m) \leq f_n^{(k)}(u)}$
dove $u = \max(x_1, \dots, x_m)$.

Mostriamo che $\mathcal{L}_n$ contiene qualche funzione che non è calcolabile mediante programmi di L_{n-1}.
PROPOSIZIONE: per $n \geq 1$, $\mathbf{f_n \in \mathcal{L}_n}$.
DIM (per induzione su n): per $n = 1$, $f_1(0) = 1$ e $f_1(x) = 2x$ per $x > 0$.
f_1 è calcolata dal seguente programma:

```
Y ← Y+1                    (questo per avere  in uscita 1 quando X=0)
LOOP X
X ← X+1
Y ← X
END
```

Assumiamo che il risultato sia vero per $n = k$, cioè $f_k \in \mathcal{L}_k$, e dimostriamolo per $n = k+1$.
Per definizione $f_{k+1}(x) = f_k^{(x)}(1)$ quindi un programma che calcoli f_{k+1} è il seguente:

```
Y ← Y+1
Z ← Z+1
LOOP X
Y ← f_k(Z)
Z ← Y
END
```

Poiché $f_k \in \mathcal{L}_k$ e abbiamo aggiunto un nuovo ciclo, allora $f_{k+1} \in \mathcal{L}_{k+1}$.

Ora dobbiamo mostrare che $f_n \notin \mathcal{L}_{n-1}$.
Diremo che un predicato è vero <u>quasi ovunque</u> quando è vero tranne che su un insieme finito di numeri. Allora <u>f è una maggiorante di g, cioè $f \succ g$, se $f(x) > g(x)$ quasi ovunque</u>.

PROPOSIZIONE: $\mathbf{f_{n+1} \succ f_n^{(k)}}$ per ogni n e k.
DIM (per induzione su n e su k): Per $n = 0$ si ha:
$f_0^{(k)}(x) = x+2+ \dots +2$ (k volte) $= x + 2k$ per $x \geq 2$
$f_1(x) = 2x$
e quindi è vera perché $2x \succ x+2k$ essendo $x > 2k$, per ogni k fissato, da un certo punto in poi.
Per $n \neq 0$ e per $k = 0$ si ha:
$f_{n+1}(x) > x$ per il lemma 3
$x = f_n^{(0)}(x)$ per definizione di $f^{(0)}$
Ora supponiamo che $f_{n+1} \succ f_n^{(k)}$ e dimostriamo che anche $f_{n+1} \succ f_n^{(k+1)}$.
Allora da un certo punto in poi avremo che :
$f_n^{(k+1)}(x) < f_n^{(k+1)}(2x\text{-}4)$ per il lemma 4 (da un certo punti in poi)
$f_n^{(k+1)}(2x\text{-}4) = f_n^{(k+1)}(f_1(x\text{-}2))$ per definizione di f_1
$f_n^{(k+1)}(f_1(x\text{-}2)) \leq f_n^{(k+1)}(f_n(x\text{-}2))$ per il lemma 4 e 5
$f_n^{(k+1)}(f_n(x\text{-}2)) = f_n^{(2)}(f_n^{(k)}(x\text{-}2))$
$f_n^{(2)}(f_n^{(k)}(x\text{-}2)) \leq f_n^{(2)}(f_{n+1}(x\text{-}2))$ per l'ipotesi induttiva (da un certo punto in poi)
$f_n^{(2)}(f_{n+1}(x\text{-}2)) = f_n^{(2)}(f_n^{(x-2)}(1)) = f_n^{(x)}(1) = f_{n+1}(x)$
Quindi $f_n^{(k+1)}(x) < f_{n+1}(x)$ quasi ovunque.

PROPOSIZIONE: $\mathbf{f_{n+1} \in \mathcal{L}_{n+1}}$, ma $\mathbf{f_{n+1} \notin \mathcal{L}_n}$.
DIM (per assurdo): Supponiamo che $f_{n+1} \in \mathcal{L}_n$. Allora si ha che:
$f_{n+1}(x) \leq f_n^{(k)}(x)$ per qualche k e per ogni x — per il corollario del teorema della limitazione
$f_n^{(k)}(x) < f_{n+1}(x)$ quasi ovunque — per la proposizione appena mostrata
$f_{n+1}(x) < f_{n+1}(x)$ quasi ovunque — contraddizione
quindi <u>non</u> possiamo assumere che $f_{n+1} \in \mathcal{L}_n$.

Con questi risultati abbiamo dimostrato che:
1. $\mathcal{L}_n$ è una gerarchia, cioè aumentando la profondità di nidificazione riusciamo a calcolare effettivamente più cose.
2. Tutte le f_n sono ricorsive primitive.