A set $\{p^j \mid 1 \le j \le n\}$ of probability distributions is a <u>mixed Nash equilibrium</u> of the load balancing game iff for all jobs t_j for all machines $s_i, s_k \in S_j$ there holds

$$p_i^j > 0 \Rightarrow L_i + (1-p_i^j)\,w_j \le L_k + (1-p_k^j)\,w_j.$$

<u>Observation:</u>

1) There holds

$$L_i + (1-p_i^j)\,w_j = \left(\sum_{l=1}^{n} p_i^l w_l\right) + (1-p_i^j)\,w_j$$

$$= \left(\sum_{\substack{l=1\\ l\neq j}}^{n} p_i^l w_l\right) + w_j$$

$\Rightarrow$

The expression $L_i + (1-p_i^j)\,w_j$ does not depend on p^j. This implies that the probability distribution of job t_j has no influence on the expected load of a server s_i under the condition that s_i executes the job t_j.

2) Each pure Nash equilibrium of the load balancing game is also a mixed Nash equilibrium where for each job exactly one probability is equal 1 and all other probabilities are 0.

First we shall consider the following criterions of quality:

1. $\sum_{j=1}^{n} c_j$ average social cost

2. $\max\{c_j \mid 1 \le j \le n\}$ fairness

With respect to fairness, Theorem 1.3 shows for the special case $S_j = S$ for $1 \le j \le n$ that for an arbitrary pure Nash equilibrium $\max\{c_j \mid 1 \le j \le n\}$ is bounded by $2 \cdot$ fairness of an optimal solution.

Goal:
The comparision of mixed Nash equilibria with optimal pure assignments.

Theorem 6.1

Let $P = \{p^j \mid 1 \le j \le n\}$ be any mixed Nash equilibrium for the load balancing game, c_j^*, $1 \le j \le n$ be with respect to P the expected load of that machine which executes the job t_j and let $S_j = S$, $1 \le j \le n$. Let opt be the minimal cost of a pure strategy with respect to the criterion $\max\{c_j \mid 1 \le j \le n\}$. Then the following is fulfilled:

$$\max\{c_j^* \mid 1 \le j \le n\} \le 2 \cdot \text{opt}.$$

Proof:

Each job is executed by a server. Hence,

$$\text{opt} \geq \max \{ w_j \mid 1 \leq j \leq n \}$$

Furthermore, each assignment has to contain a server with load $\geq \frac{1}{m} \sum_{k=1}^{n} w_k$. Hence,

$$\text{opt} \geq \frac{1}{m} \cdot \sum_{k=1}^{n} w_k .$$

For $1 \leq j \leq n$ there holds by definition

$$(+) \qquad c_j^* = \sum_{i=1}^{m} p_i^j \left(L_i + (1 - p_i^j) w_j \right)$$

For $p_i^j > 0$ the characterization of a Nash equilibrium implies

$$(++) \qquad L_i + (1-p_i^j) w_j \begin{cases} = L_k + (1-p_k^j) w_j & \text{if } p_k^j > 0 \\ \leq L_k + (1-p_k^j) w_j & \text{if } p_k^j = 0 \end{cases}$$

If we insert (++) in (+) then we obtain

$$c_j^* = L_i + (1 - p_i^j) w_j \qquad \forall i \text{ with } p_i^j > 0$$

$$(+++) \qquad = w_j + \sum_{\substack{e=1 \\ e \neq j}}^{n} p_i^e w_e$$

After summation on all servers $s \in S$ we obtain applying (++) and (+++)

$$m \cdot c_j^* \leq \sum_{s=1}^{m} \left(L_s + (1 - p_s^j) w_j \right)$$

$$= \sum_{s=1}^{m} \left(w_j + \sum_{\substack{e=1 \\ e \neq j}}^{n} p_s^e w_e \right)$$

$$= m \cdot w_j + \sum_{s=1}^{m} \sum_{\substack{e=1 \\ e \neq j}}^{n} p_s^e w_e$$

Since $w_j = \sum_{s=1}^{m} p_s^j w_j$ we obtain

$$= (m-1) w_j + \sum_{k=1}^{n} w_k$$

$$\leq m \cdot w_j + \sum_{k=1}^{n} w_k$$

$$\Longleftrightarrow \quad c_j^* \leq w_j + \frac{1}{m} \sum_{k=1}^{n} w_k$$

$$\leq 2 \cdot opt$$

Since this inequality is obtained for arbitrary j, it follows

$$\max \{ c_j^* \mid 1 \leq j \leq n \} \leq 2 \cdot opt.$$

∎

Let X_j, $1 \leq j \leq n$ be that random variable which denotes in our experiment the load of that server which executes the job t_j. Then there holds

$$c_j = E(X_j).$$

We are interested in the following third quality criterion:

3. $E(\max\{X_j \mid 1 \le j \le n\})$.

Note that $E(\max\{X_j \mid 1 \le j \le n\})$ and $\max\{E(X_j) \mid 1 \le j \le n\}$ are not the same

Goal:

An analysis of the coordination ratio

$$\frac{E(\max\{X_j \mid 1 \le j \le n\})}{opt.}$$

Theorem 6.2

The coordination ratio $\frac{E(\max\{X_j \mid 1 \le j \le n\}}{opt}$ of the load balancing game with m servers is $\Omega\left(\frac{\log m}{\log\log m}\right)$.

Proof:

Consider m jobs $t_1, t_2, \dots, t_m$ with $w_j = 1$ for $1 \le j \le m$.

Exercise:

Show that $P := \{p^j \mid 1 \le j \le m\}$ with $p_i^j = \frac{1}{m}$ for $i,j \in \{1, 2, \dots, m\}$ is a mixed Nash equilibrium.

An optimal pure strategy assigns different jobs to different servers such that all servers have load one.

The expected load of the mixed Nash equilibrium above is equal to the expected maximal number of balls in a box if we have m boxes and each ball is placed into one of the m boxes with probability $\frac{1}{m}$ for all boxes. This expected maximal number of balls is

$$\Theta\left(\frac{\log m}{\log \log m}\right).$$

The upper bound is relatively easy to prove. See

<u>Kurt Mehlhorn</u>, Data Structures and Algorithms 1: Sorting and Searching, Springer 1984, p. 123 - 124.

We need the lower bound which is much harder to prove. See

<u>Gaston H. Gonnet</u>, Expected Length of the Longest Probe Sequence in Hash Code Searching, JACM 28 (1981), 289 - 304.

∎

Our goal is to prove that the lower bound above is ~~sharp.~~ tight.

For doing this, we need the following standard version of the Hoeffding-bound.

Hoeffding-bound

Let $X_1, X_2, \dots, X_N$ be N independent random variables with values in the interval $[0, z]$ for a $z > 0$. Let $X = \sum_{i=1}^{N} X_i$. Then for each t the following is fulfilled

$$\Pr\left[\sum_{i=1}^{N} X_i \geq t\right] \leq \left(\frac{e \cdot E[X]}{t}\right)^{t/z}$$

See

Wassily Hoeffding, Probability inequalities for sums of bounded random variables, Journal of American Statistical Association 58 (1963), 13-30.

Theorem 6.3

The coordination ratio $\frac{E(\max\{X_i \mid 1 \leq i \leq n\})}{opt}$ for the load balancing game with m servers is $O\left(\frac{\log m}{\log \log m}\right)$.

Proof:

W.l.o.g. we can assume $opt = 1$.

(By scaling the weights. This means by division by the social optimum)

Since $opt \geq \max\{w_j \mid 1 \leq j \leq n\}$ it holds

$$w_j \leq 1 \quad \text{for } 1 \leq j \leq n.$$

We need the following simple implication of the characterization of a mixed Nash equilibrium:

Lemma 6.1

Let $P = \{p^j \mid 1 \leq j \leq n\}$ be a mixed Nash equilibrium. Then for all $1 \leq i \leq n$ and all $1 \leq q \leq m$ the following is fulfilled:

i) $p_q^i > 0 \Rightarrow L_j + w_i \geq L_q \quad \forall j$

ii) $(p_j^i > 0 \text{ and } p_q^i > 0) \Rightarrow |L_j - L_q| \leq opt.$

Proof:

i) characterization of a mixed Nash equilibrium $\Rightarrow$

$$p_q^i > 0 \Rightarrow L_q + (1 - p_q^i) w_i \leq L_j + (1 - p_j^i) w_i$$

After the addition of both sides by $p_j w_i$ we obtain

$$L_q + \underbrace{(1 - p_q^i + p_j^i) w_i}_{\geq 0} \leq L_j + w_i$$

$$\Rightarrow \quad L_q \leq L_j + w_i$$

ii) Since $w_i \leq opt$ the assertion ii) follows from i).

$\square$

Lemma 6.2

Let $P = \{ p^j \mid 1 \le j \le n \}$ be a Nash equilibrium. Then

$$c = \max \{ L_i \mid 1 \le i \le m \} < 2.$$

Proof:

Assume $\max \{ L_i \mid 1 \le i \le m \} \geq 2$.

$\Rightarrow$

$\exists\, i_0$ with $L_{i_0} \geq 2$.

Consider any job t_j with $p^j_{i_0} > 0$. Since $L_{i_0} > 0$ such a job exists.

Lemma 6.1. i) $\Rightarrow$

$$\forall\, 1 \le i \le m : \quad L_i + w_j \geq L_{i_0} \geq 2$$

Hence

$$\sum_{i=1}^{m} L_i + (m-1)\, w_j \geq 2 \cdot m$$

$$\Leftrightarrow \sum_{e=1}^{n} w_e + (m-1)\, w_j \geq 2m$$

$$\Leftrightarrow \frac{1}{m} \cdot \sum_{e=1}^{n} w_e + \underbrace{\frac{(m-1)}{m} \cdot w_j}_{<1} \geq 2$$

$$\Rightarrow \qquad \frac{1}{m} \cdot \sum_{e=1}^{n} w_e > 1$$

This is a contradiction since

$$1 = \mathrm{opt} \geq \frac{1}{m} \sum_{e=1}^{n} w_e > 1.$$

$\square$

For $1 \leq i \leq m$, $1 \leq j \leq n$ let

- C_i be the random variable for the load of server s_i and
- D_i^j be the 0-1 random variable with $Pr[D_i^j = 1] = p_i^j$

Then

$$C_i = \sum_{j=1}^{n} D_i^j w_j$$

$D_i^1 w_1, D_i^2 w_2, \ldots, D_i^n w_n$ are n independent random variables with values in the interval $[0,1]$.

$\Rightarrow$

We can apply the Hoeffding bound.

$\rightsquigarrow$

$$Pr[C_i \geq t] \leq \left(\frac{e \cdot E(C_i)}{t}\right)^t$$

$$\leq \left(\frac{2 \cdot e}{t}\right)^t$$

$$(E(C_i) = L_i \text{ and } L_i < 2)$$

Note that for $t_0 \geq \frac{3 \ln m}{\ln \ln m}$ there holds

$$Pr[C_i \geq t_0] \ll \frac{1}{m}$$

This leads to

$$E[\max\{C_i \mid 1 \leq i \leq m\}] = O(t_0)$$

More exactly, we obtain for t_0:

$$E[\max\{C_i \mid 1 \le i \le m\}]$$

$$\le t_0 + \sum_{J=t_0}^{\infty} \Pr[\exists j \in \{1,2,\dots,m\} : C_j \ge J]$$

$$\le t_0 + \sum_{J=t_0}^{\infty} m \cdot \left(\frac{2e}{J}\right)^J$$

$$\le t_0 + \sum_{J=t_0}^{\infty} 2^{-J}$$

$$\le t_0 + 1$$

$\blacksquare$

The paper

Petra Berenbrink, Leslie Ann Goldberg, Paul Goldberg, Russell Martin, Utilitarian Resource Assignment, Journal of Discrete Algorithms 4 (2006), 567-587. (can be obtained from the home page of Paul Goldberg).

considers the coordination ratio with respect to the average social cost.

Artur Czumaj, Selfish Routing on the Internet, in J. Leung (ed.), Handbook of Scheduling: Algorithms, Models, and Performance Analysis, CRC Press 2004 (can be obtained from the homepage of Artur Czumaj).

gives an overview.

7. Load balancing games with partial central control

Reference:

Tim Roughgarden, Stackelberg Scheduling Strategies, SIAM J. Comput. 33 (2004), 332 - 350.

Considering the load balancing game, we shall investigate the following question:

Can a system of noncooperating selfish users be improved by a central control of "some" users where the goal of the central control is to optimize the whole system?

Such problems can be formulized by an optimization problem with respect to Stackelberg-games. Heinrich von Stackelberg has developed his concept for two-persons-games. We shall generalize this concept to n-persons-games.

A Stackelberg-game is an n-persons-game with one player a leader. First the leader fixed his strategy while the other players (the followers) react independently and selfishly to the leader's strategy. A Nash equilibrium relative to the leader's strategy is called a Stackelberg equilibrium.

We modify the load balancing game obtaining the corresponding Stackelberg game as follows:

The model (using the notation of Roughgarden's paper):

- $M = \{1, 2, \dots, m\}$ m servers (or machines)
  For each server $i \in M$ there is a nonnegative, continuous and nondecreasing latency function $\ell_i : \mathbb{R}^+ \to \mathbb{R}^+$. Furthermore, $x_i \cdot \ell_i(x_i)$ is a convex function. x_i denotes the load of the server i.

- We assume a finite and positive rate r of job arrivals. An assignment of the jobs to the machines is an m-vector $x \in \mathbb{R}_+^m$ such that $\sum_{i=1}^m x_i = r$.
  For a subset $M' \subseteq M$ of the machines, we write $x(M') := \sum_{i \in M'} x_i$.

- The cost or total latency $C(x)$ of an assignment x is defined by
$$C(x) := \sum_{i=1}^{m} x_i \cdot \ell_i(x_i).$$

- (M, r) denotes an instance with servers M, rate r, and no central control.
  An instance with centrally controlled jobs (a Stackelberg instance) is denoted by

(M, r, α), where the third parameter $\alpha \in (0,1)$ indicates the fraction of the overall traffic that is centrally controlled.

In a Stackelberg instance the goal of the central control is to distribute his fraction of r in such a way to the machines that the total latency $C(x)$ is minimized in an equilibrium. The following questions suggest themselves:

1) Given a set of machines and a set of jobs is it possible to characterize and to compute within all possible strategies of the leader a strategy which results into an equilibrium?

2) What is the worst case ratio of the total latency in a Stackelberg equilibrium and an optimal assignment?

First, we need some definitions.

A <u>Stackelberg strategy</u> for the Stackelberg instance (M, r, α) is an assignment feasible for $(M, \alpha r)$. Let s be a strategy for Stackelberg instance (M, r, α) where machine $i \in M$ has latency function ℓ_i, and let $\tilde{\ell}_i(x_i) := \ell_i(s_i + x_i)$ for each $i \in M$. An <u>equilibrium induced by strategy s</u> is a Nash equilibrium for $(M, (1-\alpha)r)$ with respect to the latency functions $\tilde{\ell} = (\tilde{\ell}_1, \tilde{\ell}_2, \dots, \tilde{\ell}_m)$.

Note that the definition above reduces Stackelberg equilibria to Nash equilibria with respect to a corresponding game without central control. The load balancing game is a special case of the nonatomic routing game. Hence, the theorems proved in Chapter 2 can be applied such that the following lemma can be obtained:

<u>Lemma 7.1</u>

Let s be a strategy for a Stackelberg instance with continuous, nondecreasing latency functions. Then there exists an assignment induced by s. Furthermore, all assignments induced by s have equal cost.

<u>Proof:</u>

Exercise

■

The following simple observation will be useful for the development of algorithms for the computation of good Stackelberg strategies.

<u>Lemma 7.2</u>

Let s be a strategy for Stackelberg instance (M, r, α). Let $t = (t_1, t_2, \dots, t_m)$ be an equilibrium induced by s. Let M' denote the machines on which $t_i > 0$; i.e., $M' := \{ i \in M \mid t_i > 0 \}$. Let $s(M') = \sum_{i \in M'} s_i$, $t(M') := \sum_{i \in M'} t_i$ and $(s+t)' = (s_{i_1} + t_{i_1}, s_{i_2} + t_{i_2}, \dots, s_{i_{m'}} + t_{i_{m'}})$

where $M' = \{i_1, i_2, \dots, i_{m'}\}$. Then $(s+t)'$ is a Nash equilibrium for $(M', s(M') + t(M'))$. In particular, all machines in M' have the same latency with respect to $s+t$.

Proof:

Exercise

■

Example 7.1

a) $M = \{1,2\}$, $\ell_1(x_1) = 1$, $\ell_2(x_2) = x_2$, $r = 1$

optimal assignment:

$(\frac{1}{2}, \frac{1}{2})$ total latency $3/4$

Nash equilibrium:

$(0,1)$ total latency 1

Consider $\alpha = \frac{1}{2}$ and the Stackelberg strategy $s = (\frac{1}{2}, 0)$.

$\Rightarrow$

The equilibrium $(0, \frac{1}{2})$ induced by s results into an optimal assignment of all jobs.

b) $M = \{1,2\}$, $\ell_1(x_1) = 1$, $\ell_2(x_2) = 2 \cdot x_2$, $r = 1$

optimal assignment:

$(\frac{3}{4}, \frac{1}{4})$ total latency $\frac{7}{8}$

Nash equilibrium:

$(\frac{1}{2}, \frac{1}{2})$ total latency 1

Consider $\alpha = \frac{1}{2}$.

<u>Exercise</u>:

Prove that each Stackelberg strategy s results into the equilibrium $(\frac{1}{2}, \frac{1}{2})$ induced by s.

The exercise above shows that with respect to $\alpha = \frac{1}{2}$ no Stackelberg strategy results into an optimal assignment.

♦

In the paper above, Tim Roughgarden proves that the computation of an optimal Stackelberg strategy is NP-hard. Hence, we will investigate algorithms for the computation of "almost optimal" Stackelberg strategies.

Let us consider the example above again. The optimal assignments assigns load to the machine which has a realitively high latency function. But selfish players avoid such machines. Therefore, a good Stackelberg strategy should favour such machines. This observation suggests the following heuristic:

LLF (largest latency first)

(1) Compute the optimal assignment x^* for (M, r)

(2) Index the machines of M so that

$$\ell_1(x_1^*) \leq \ell_2(x_2^*) \leq \dots \leq \ell_m(x_m^*).$$

(3) Let $k \leq m$ be minimal with

$$\sum_{j=k+1}^{m} x_j^* \leq \alpha \cdot r.$$

(4) Define $s = (s_1, s_2, \dots, s_m)$ by

$$s_i := \begin{cases} 0 & \text{if } i < k \\ \alpha r - \sum_{j=k+1}^{m} x_j^* & \text{if } i = k \\ x_i^* & \text{if } i > k. \end{cases}$$

- Since the load balancing game is a special case of the nonatomic routing game, it follows directly that LLF can be computed in polynomial time.
- Tim Roughgarden has shown that for linear latency functions LLF can be computed in $O(m^2)$ time.

Goal:

Analysis of LLF for arbitrary sets of latency functions and an arbitrary number of machines.

This means we shall investigate the question what is the worst case ratio between the total latency of Stackelberg equilibria induced by LLF and of an optimal assignment.

The following example shows that this worst case ratio is at least $\frac{1}{\alpha}$.

<u>Example 7.2</u>

We vary Example 7.1. Let

$M = \{1,2\}$, $\ell_1(x_1) = 1$, $\ell_2(x_2) = 2^k \cdot x_2^k$, $r = 1$ for $k \in \mathbb{N}$. Furthermore, let $\alpha = \frac{1}{2}$.

Any Stackelberg strategy induces the assignment $(\frac{1}{2}, \frac{1}{2})$ having total latency 1.

The optimal assignment is $(\frac{1}{2} + \delta_k, \frac{1}{2} - \delta_k)$ having cost $\frac{1}{2} + \varepsilon_k$, where $\delta_k, \varepsilon_k \to 0$ as $k \to \infty$.

$\Rightarrow$

The best induced assignment may be (arbitrarily close to) twice as costly as the optimal assignment.

<u>Exercise:</u>

For $\alpha \in (0,1)$ construct an example which shows that the best assignment induced by a Stackelberg strategy may be $\frac{1}{\alpha}$ times as costly as the optimal assignment.

We shall prove that the LLF heuristic always induces an assignment which is at most $\frac{1}{\alpha}$ times as costly as the optimal assignment.

Idea:

Exploit the iterative structure of the LLF strategy and proceed by induction on the number of machines. If the LLF strategy first saturates the m-th machine then apply the inductive hypothesis to the remainder of the LLF strategy on the first m-1 machines to derive a performance guarantee.

This idea nearly succeeds, but there are two difficulties.

1) It is possible that the LLF strategy does not saturate any machine.

2) In order to obtain a clean application of the inductive hypothesis to the first m-1 machines, we require that the optimal and LLF-induced assignments place the same total amount of jobs on these machines; i.e., the equilibrium induced by the LLF strategy assigns to the m-th machine only those jobs, which are already assigned to the m-th machine by the LLF strategy.

The following lemma solves the second problem.

Lemma 7.3

Let (M, r, α) be a Stackelberg instance with optimal assignment x^*. Index the machines of M so that $\ell_m(x_m^*) \geq \ell_i(x_i^*)$ for $1 \leq i \leq m$. If s is a strategy with $s_m = x_m^*$, then there exists an induced equilibrium t with $t_m = 0$.

Remark:

Lemma 7.1 $\Rightarrow$

All equilibria induced by s have the same cost. Hence, Lemma 7.3 is sufficient for our purpose.

Proof:

Let t be an equilibrium induced by s with t_m minimal.

Assume that $t_m > 0$. Let

$$L := \ell_m(s_m + t_m) = \ell_m(x_m^* + t_m)$$

be the latency having all machines i with $t_i > 0$ with respect to $s+t$.

Lemma 7.2 $\Rightarrow$ L is well defined.

Assume that $\ell_m(x_m^*) < L$

$$\left(\ell_m(x_m^*) \geq \ell_i(x_i^*) \text{ for all } i \right) \Rightarrow$$

$$\ell_i(x_i^*) < L \quad \text{for } 1 \leq i \leq m.$$

By construction we obtain

$$s_i \le x_i^* \quad \text{for } 1 \le i \le m$$

If $\exists\, i \in \{1, 2, \dots, m-1\}$ with $s_i < x_i^*$ then a portion >0 can be placed from the m-th machine to the ith machine without increasing the total latency.

A contradiction to the choice of t.

$\Rightarrow$ $$s_i = x_i^* \quad \text{for } 1 \le i \le m-1.$$

x^* and $s+t$ are assignments of the same rate. Hence, $t_m = 0$

a contradiction.

Therefore $\ell_m(x_m^*) \geq L$.

ℓ_m nondecreasing $\Rightarrow$

$$\ell_m(x) = L \quad \forall\, x \in [x_m^*,\, x_m^* + t_m].$$

Consider j with $s_j + t_j < x_j^*$

Because of $s_m + t_m > x_m^*$ there exists such a j.

Note that

$$\ell_j(s_j + t_j) \le \ell_j(x_j^*) \le \ell_m(x_m^*) = L$$

Since ℓ_j is nondecreasing this implies

$$\ell_j(x) \le L \quad \forall\, x \in [x_j + t_j,\, x_j^*].$$

Hence, a portion > 0 can be placed from the m-th machine to the j-th machine without increasing the total latency.

A contradiction to the choice of t.

Therefore, the assumption $t_m > 0$ is wrong so that we have proved the lemma.

∎

Now we can prove the desired result.

Theorem 7.1

Let (M, r, α) be a Stackelberg instance, x^* be an optimal assignment for (M, r), s be the corresponding LLF strategy and t be an equilibrium induced by s. Then

$$C(s+t) \leq \frac{1}{\alpha} \cdot C(x^*).$$

Proof: (by induction on the number m of machines)

$m = 1$.

For one machine, the assertion is fulfilled for all ℓ, r and α. ✓

Let $m > 1$.

Inductive hypothesis:

For all $m' < m$, the assertion is fulfilled for all ℓ, r and α.

Let (M, r, α) be an arbitrary Stackelberg instance with m machines.

Let x^* be an optimal assignment for (M, r). We order the machines such that

$$\ell_1(x_1^*) \leq \ell_2(x_2^*) \leq \ldots \leq \ell_m(x_m^*).$$

Let s be the corresponding LLF strategy

W.l.o.g., we can assume that $r = 1$ (otherwise, use the latency functions $\tilde{\ell}$ with $\tilde{\ell}_i(x_i) := \ell_i(r \cdot x_i)$)

Let L be that latency having all machines i with $t_i > 0$ with respect to $s+t$.

Lemma 7.2 $\Rightarrow$ L is well defined.

We distinguish two cases.

<u>Case 1</u>: $\exists\, i \in \{1, 2, \ldots, m\}$ with $t_i = 0$.

Let

$M_1 := \{ i \in M \mid t_i = 0 \}$ and
$M_2 := \{ i \in M \mid t_i > 0 \}$.

For $\alpha = 1$ the theorem is trivial. $\alpha < 1$ implies that $M_2 \neq \emptyset$.

Denote by α_i, $i = 1, 2$ that portion of centrally controlled jobs which are assigned to the machines in M_i; i.e., $\alpha_i = s(M_i)$.

Let C_i denote the total latency of the machines in M_i caused by $s+t$.

Lemma 7.2 $\Rightarrow$

$$C_2 = (1-\alpha_1) L \quad \text{and} \quad C_1 \geq \alpha_1 L.$$

<u>Observation:</u>

- x^* restricted to M_2 is an optimal assignment for $(M_2, 1-\alpha_1)$
- S restricted to M_2 is a LLF strategy for $(M_2, 1-\alpha_1, \alpha')$ where $\alpha' = \frac{\alpha_2}{1-\alpha_1}$.

$\Rightarrow$

We can apply the inductive hypothesis to $(M_2, 1-\alpha_1, \alpha')$.

The application to $(M_2, 1-\alpha_1, \alpha')$ and $x_i^* \geq S_i = s_i + t_i \ \forall \ i \in M_1$ give us

$$C(x^*) \geq C_1 + \alpha' C_2$$

Hence, it suffices to prove

$$\alpha(C_1 + C_2) \leq C_1 + \alpha' C_2$$

Since $\alpha \leq 1$ and $C_1 \geq \alpha_1 L$ it suffices to prove

$$\alpha(\alpha_1 L + C_2) \leq \alpha_1 L + \alpha' C_2.$$

After substitution of $C_2 = (1-\alpha_1) L$ and $\alpha' = \frac{\alpha_2}{1-\alpha_1}$

and division by L, we obtain

$$\alpha(\alpha_1 + 1 - \alpha_1) \leq \alpha_1 + \frac{\alpha_2}{1-\alpha_1}(1-\alpha_1)$$

$$\Longleftrightarrow \qquad \alpha \leq \alpha$$

such that the Case 1 is proved.

<u>Case 2:</u> $t_i > 0 \quad \forall\, i \in \{1, 2, \ldots, m\}$.

$\Longrightarrow$

- $C(s+t) = L$ and
- $s_m < x_m^*$ (by Lemma 7.3 and $t_m > 0$)

$\Longrightarrow$
since $r = 1$ $\qquad \alpha < x_m^*$.

Assume that $l_m(x_m^*) < L$

$\Longrightarrow$

$$l_i(x_i^*) < L \quad \text{for } 1 \leq i \leq m$$

But $l_i(s_i + t_i) = L$ for $1 \leq i \leq m$

This contradicts that x^* and $s+t$ are assignments with respect to the same rate r.

Hence, $l_m(x_m^*) \geq L$

$\Longrightarrow$

$$C(x^*) \geq x_m^* \, l_m(x_m^*) \geq \alpha L = \alpha C(s+t)$$

such that the Case 2 is proved.

∎