New Heuristics and Integer Programming Formulations for Scheduling Divisible Load Tasks

Elbio R. Abib (Microsoft, USA)
Celso C. Ribeiro (UFF, Brazil)

IEEE CISched 2009, Nashville

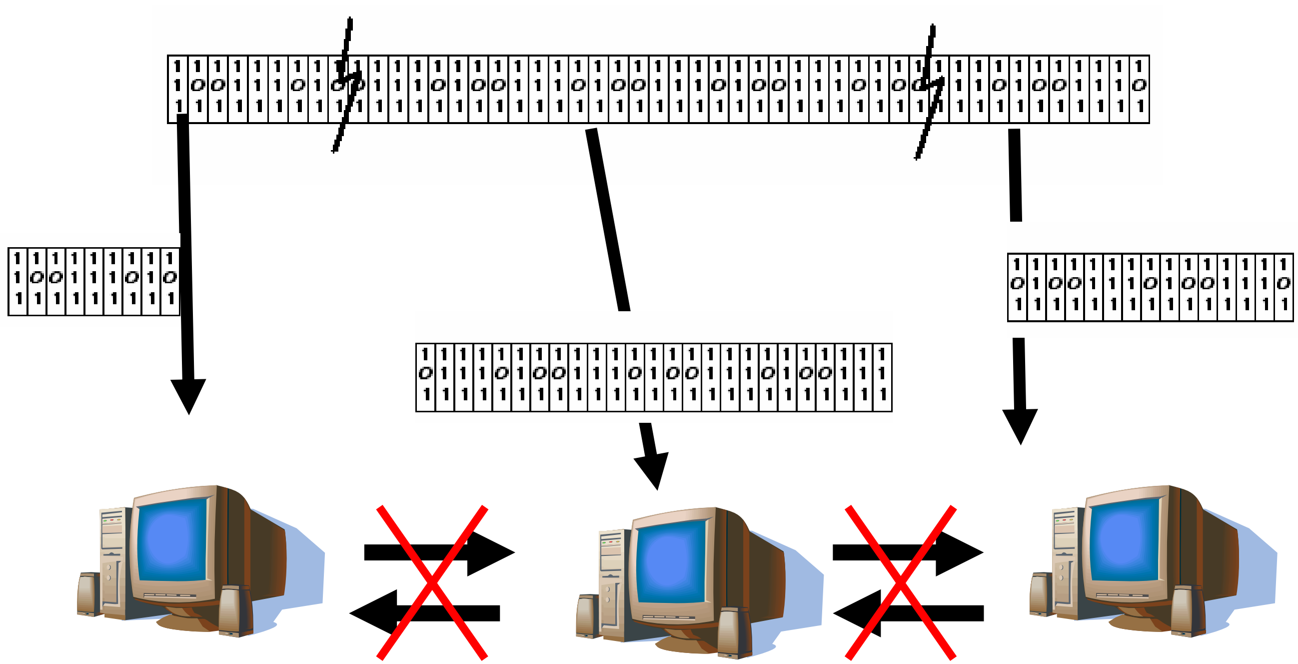
Agenda

- Divisible load model
  - System model and problem formulation
  - Single- and multi-installment scheduling
- Single-installments: mixed integer programming
- Linear-time algorithm for a given activation order
- Fast constructive heuristic with feedback
- Computational experiments
- Multiple-installments: mixed integer programming

Divisible load model

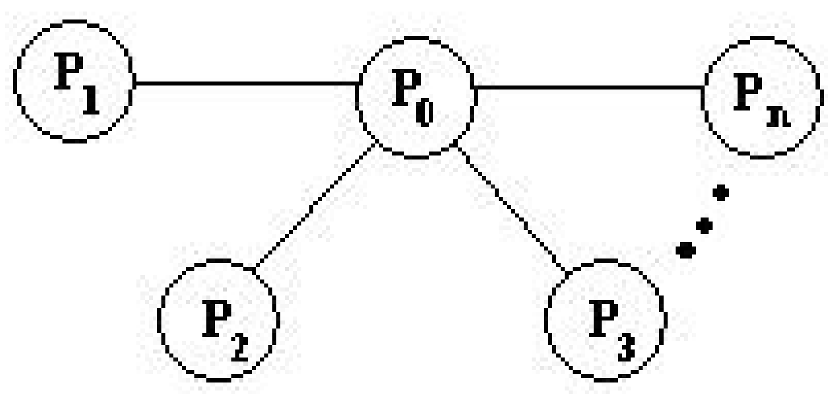
Divisible load model

- Load may be split continuously into arbitrarily many small chunks
- No precedence constraints



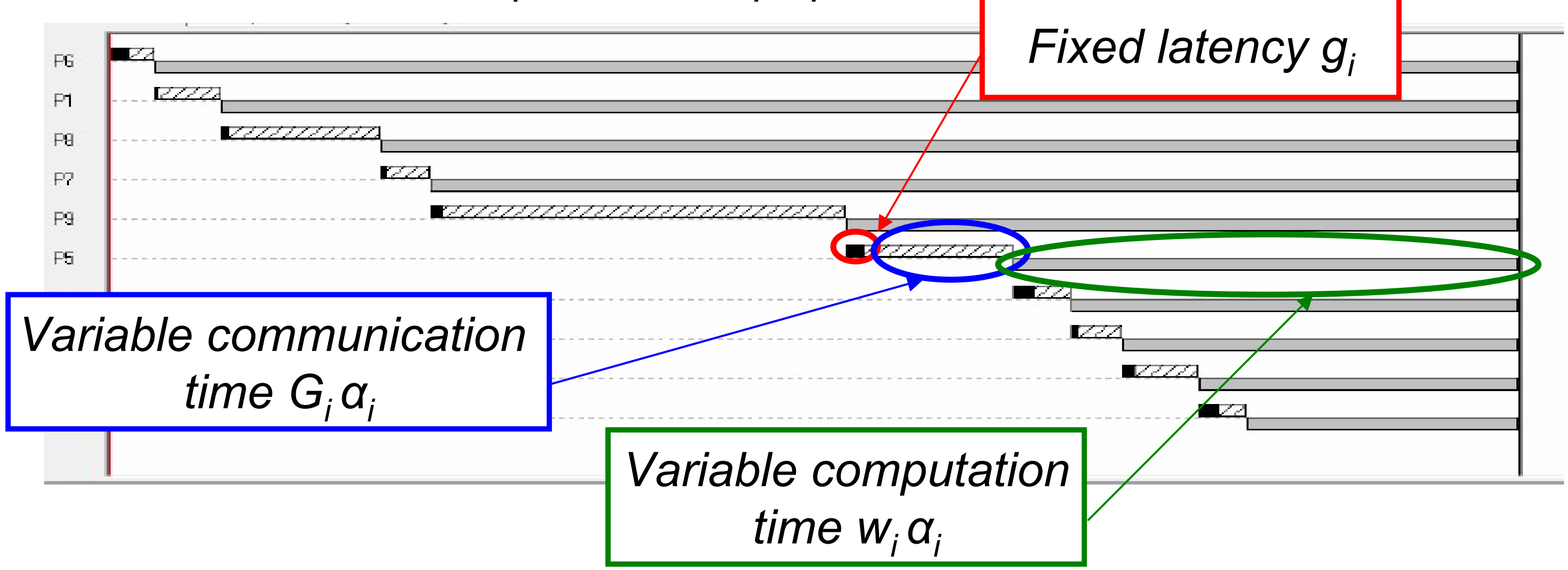
System model and problem formulation

- Interconnection topology: star network
  - Dedicated grid
- Model: one master - n workers
  - Master owns the total load W
- No communication/computation overlap in any processor
- No communication overlap through the master



System model and problem formulation

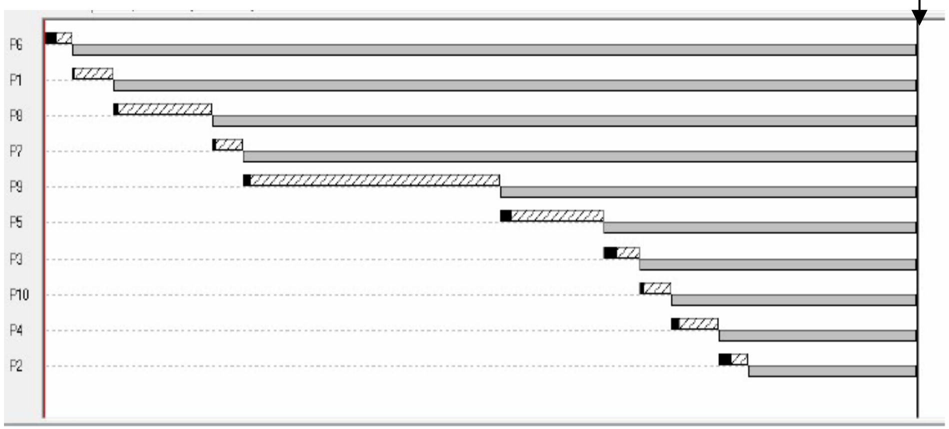
- Single-installment scheduling
  - Each processor receives portion α_i of total load
  - Master takes $g_i + G_i\alpha_i$ time units to send the data to processor P_i
  - Processor P_i takes $w_i\alpha_i$ time units to process data



Single-installment scheduling

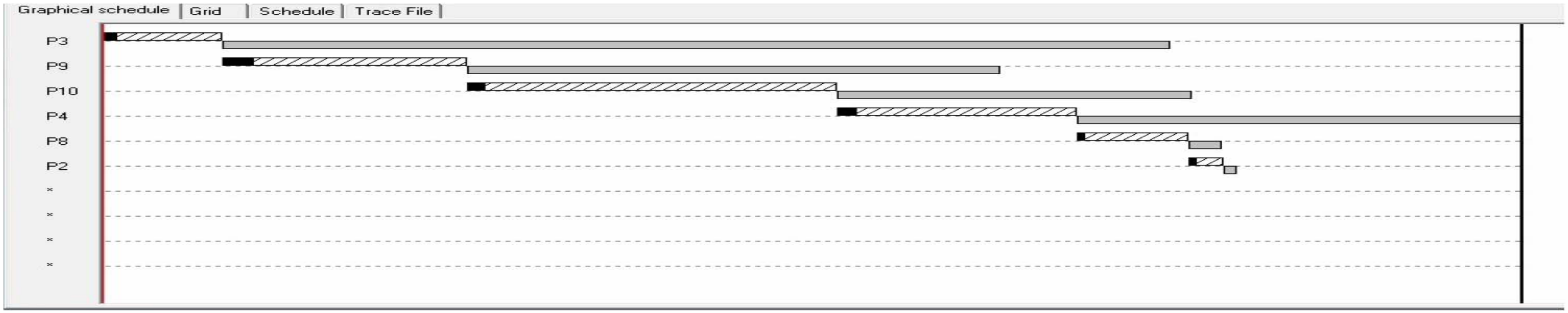
Optimal scheduling

makespan



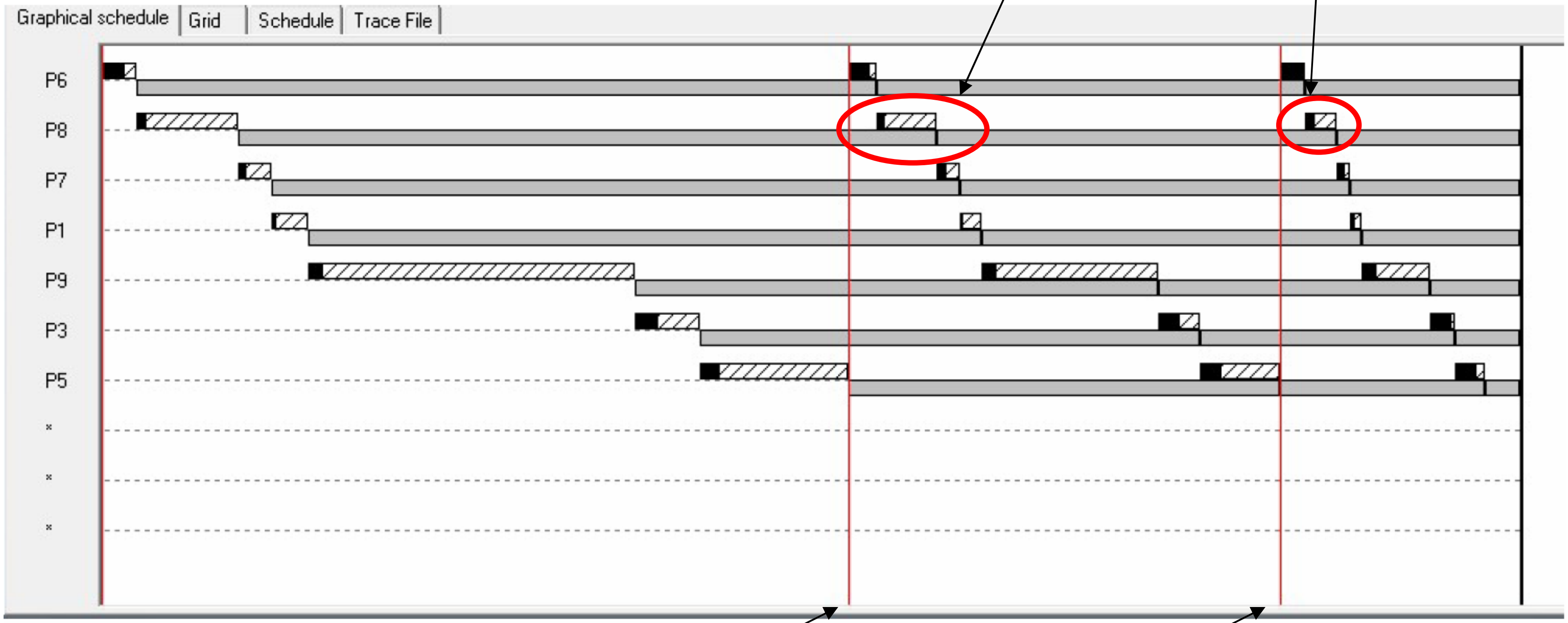
makespan

Non-optimal scheduling



Multi-installment scheduling

Communication/computation concurrency



New period

New period

Related work

- Divisible load model introduced by Cheng and Robertazzi (1988)
- Effect of latency in communication studied by Blazewicz and Drozdowski (1997)
- Beaumont et al. (2005): non-linear integer programming formulation for single-installment systems with latencies
- Linear integer programming formulations for single- and multi-installment systems with latencies not available

Single-installment mixed integer programming formulation

Single-installment scheduling

- Problem consists of determining
  - the processors to be used (and their number),
  - their activation order,
  - and their loads,
- ... so as to minimize the makespan.

Formulation

x_{ij}=1, if processor P_i is the j-th to be activated and to receive data
x_{ij}=0, otherwise

α_{ij}>0, is the amount of data sent to P_i if it is the j-th to be activated
α_{ij}=0, otherwise

t_j is the time in which the j-th processor to be activated starts receiving its data

makespan

$$T^* = \text{minimum } T \tag{1}$$

subject to:

$$\sum_{i=1}^{n} x_{ij} \leq 1 \qquad j = 1, \ldots, n \tag{2}$$

$$\sum_{j=1}^{n} x_{ij} \leq 1 \qquad i = 1, \ldots, n \tag{3}$$

$$\sum_{i=1}^{n} x_{ij} \geq \sum_{i=1}^{n} x_{i,j+1} \qquad j = 1, \ldots, n-1 \tag{4}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{ij} = W \tag{5}$$

$$\alpha_{ij} \leq W x_{ij} \qquad i, j = 1, \ldots, n \tag{6}$$

$$t_1 = 0 \tag{7}$$

$$t_j \geq t_{j-1} + \sum_{i=1}^{n} (g_i x_{i,j-1} + G_i \alpha_{i,j-1}) \qquad j = 2, \ldots, n \tag{8}$$

$$t_j + \sum_{i=1}^{n} (g_i x_{ij} + (G_i + w_i)\alpha_{ij}) = T \qquad j = 1, \ldots, n \tag{9}$$

$$x_{ij} \in \{0, 1\} \qquad i, j = 1, \ldots, n \tag{10}$$

$$\alpha_{ij} \geq 0 \qquad i, j = 1, \ldots, n. \tag{11}$$

Formulation

x_{ij}=1, if processor P_i is the j-th to be activated and to receive data
x_{ij}=0, otherwise

α_{ij}>0, is the amount of data sent to P_i if it is the j-th to be activated
α_{ij}=0, otherwise

t_j is the time in which the j-th processor to be activated starts receiving its data

$$T^* = \text{minimum } T \tag{1}$$

subject to:

$$\sum_{i=1}^{n} x_{ij} \leq 1 \qquad j = 1, \ldots, n \tag{2}$$

At most one processor can be the j-th to be activated

$$\sum_{j=1}^{n} x_{ij} \leq 1 \qquad i = 1, \ldots, n \tag{3}$$

$$\sum_{i=1}^{n} x_{ij} \geq \sum_{i=1}^{n} x_{i,j+1} \qquad j = 1, \ldots, n-1 \tag{4}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{ij} = W \tag{5}$$

$$\alpha_{ij} \leq W x_{ij} \qquad i, j = 1, \ldots, n \tag{6}$$

$$t_1 = 0 \tag{7}$$

$$t_j \geq t_{j-1} + \sum_{i=1}^{n} (g_i x_{i,j-1} + G_i \alpha_{i,j-1}) \qquad j = 2, \ldots, n \tag{8}$$

$$t_j + \sum_{i=1}^{n} (g_i x_{ij} + (G_i + w_i)\alpha_{ij}) = T \qquad j = 1, \ldots, n \tag{9}$$

$$x_{ij} \in \{0, 1\} \qquad i, j = 1, \ldots, n \tag{10}$$

$$\alpha_{ij} \geq 0 \qquad i, j = 1, \ldots, n. \tag{11}$$

Formulation

x_{ij}=1, if processor P_i is the j-th to be activated and to receive data
x_{ij}=0, otherwise

α_{ij}>0, is the amount of data sent to P_i if it is the j-th to be activated
α_{ij}=0, otherwise

t_j is the time in which the j-th processor to be activated starts receiving its data

$$T^* = \text{minimum } T \tag{1}$$

subject to:

$$\sum_{i=1}^{n} x_{ij} \leq 1$$

A processor may be activated in at most one position

$$\sum_{j=1}^{n} x_{ij} \leq 1 \qquad i = 1, \ldots, n \tag{3}$$

$$\sum_{i=1}^{n} x_{ij} \geq \sum_{i=1}^{n} x_{i,j+1} \qquad j = 1, \ldots, n-1 \tag{4}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{ij} = W \tag{5}$$

$$\alpha_{ij} \leq W x_{ij} \qquad i, j = 1, \ldots, n \tag{6}$$

$$t_1 = 0 \tag{7}$$

$$t_j \geq t_{j-1} + \sum_{i=1}^{n} (g_i x_{i,j-1} + G_i \alpha_{i,j-1}) \qquad j = 2, \ldots, n \tag{8}$$

$$t_j + \sum_{i=1}^{n} (g_i x_{ij} + (G_i + w_i)\alpha_{ij}) = T \qquad j = 1, \ldots, n \tag{9}$$

$$x_{ij} \in \{0, 1\} \qquad i, j = 1, \ldots, n \tag{10}$$

$$\alpha_{ij} \geq 0 \qquad i, j = 1, \ldots, n. \tag{11}$$

Formulation

x_{ij}=1, if processor P_i is the j-th to be activated and to receive data
x_{ij}=0, otherwise

α_{ij}>0, is the amount of data sent to P_i if it is the j-th to be activated
α_{ij}=0, otherwise

t_j is the time in which the j-th processor to be activated starts receiving its data

$$T^* = \text{minimum } T \quad (1)$$

subject to:

$$\sum_{i=1}^{n} x_{ij} \leq 1$$

$$\sum_{j=1}^{n} x_{ij} \leq 1$$

A processor may be the (j+1)-th to be activated only if there are other j processors already activated

$$\sum_{i=1}^{n} x_{ij} \geq \sum_{i=1}^{n} x_{i,j+1} \qquad j = 1, \ldots, n-1 \quad (4)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{ij} = W \quad (5)$$

$$\alpha_{ij} \leq W x_{ij} \qquad i, j = 1, \ldots, n \quad (6)$$

$$t_1 = 0 \quad (7)$$

$$t_j \geq t_{j-1} + \sum_{i=1}^{n} (g_i x_{i,j-1} + G_i \alpha_{i,j-1}) \qquad j = 2, \ldots, n \quad (8)$$

$$t_j + \sum_{i=1}^{n} (g_i x_{ij} + (G_i + w_i)\alpha_{ij}) = T \qquad j = 1, \ldots, n \quad (9)$$

$$x_{ij} \in \{0, 1\} \qquad i, j = 1, \ldots, n \quad (10)$$

$$\alpha_{ij} \geq 0 \qquad i, j = 1, \ldots, n. \quad (11)$$

Formulation

$x_{ij}=1$, if processor P_i is the j-th to be activated and to receive data
$x_{ij}=0$, otherwise

$\alpha_{ij}>0$, is the amount of data sent to P_i if it is the j-th to be activated
$\alpha_{ij}=0$, otherwise

t_j is the time in which the j-th processor to be activated starts receiving its data

$$T^* = \text{minimum } T \tag{1}$$

subject to:

$$\sum_{i=1}^{n} x_{ij} \leq 1 \qquad j = 1, \ldots, n \tag{2}$$

$$\sum_{j=1}^{n} x_{ij} \leq 1 \qquad i = 1, \ldots, n \tag{3}$$

$$\sum_{i=1}^{n} x_{ij} \geq \sum_{i=1}^{n} x_{i,j+1} \qquad j = 1, \ldots, n-1 \tag{4}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{ij} = W \tag{5}$$

Total load W has to be processed

$$\alpha_{ij} \leq W x_{ij}$$

$$t_1 = 0$$

$$t_j \geq t_{j-1} + \sum_{i=1}^{n} (g_i x_{i,j-1} + G_i \alpha_{i,j-1}) \qquad j = 2, \ldots, n \tag{8}$$

$$t_j + \sum_{i=1}^{n} (g_i x_{ij} + (G_i + w_i)\alpha_{ij}) = T \qquad j = 1, \ldots, n \tag{9}$$

$$x_{ij} \in \{0, 1\} \qquad i, j = 1, \ldots, n \tag{10}$$

$$\alpha_{ij} \geq 0 \qquad i, j = 1, \ldots, n. \tag{11}$$

Formulation

x_{ij}=1, if processor P_i is the j-th to be activated and to receive data
x_{ij}=0, otherwise

α_{ij}>0, is the amount of data sent to P_i if it is the j-th to be activated
α_{ij}=0, otherwise

t_j is the time in which the j-th processor to be activated starts receiving its data

$$T^* = \text{minimum } T \qquad (1)$$

subject to:

$$\sum_{i=1}^{n} x_{ij} \leq 1 \qquad j = 1, \ldots, n \qquad (2)$$

$$\sum_{j=1}^{n} x_{ij} \leq 1 \qquad i = 1, \ldots, n \qquad (3)$$

$$\sum_{i=1}^{n} x_{ij} \geq \sum_{i=1}^{n} x_{i,j+1} \qquad j = 1, \ldots, n-1 \qquad (4)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{ij} = W \qquad (5)$$

$$\alpha_{ij} \leq W x_{ij} \qquad i, j = 1, \ldots, n \qquad (6)$$

Processor i can only receive load as the j-th if it is the j-th to be activated

$$t_1 = 0$$

$$t_j \geq t_{j-1} + \sum_{i=1}^{n} (g_i x_{i,j-1} + G_i \alpha_{i,j-}$$

$$t_j + \sum_{i=1}^{n} (g_i x_{ij} + (G_i + w_i)\alpha_{ij}) = T$$

$$x_{ij} \in \{0, 1\} \qquad i, j = 1, \ldots, n \qquad (10)$$

$$\alpha_{ij} \geq 0 \qquad i, j = 1, \ldots, n. \qquad (11)$$

Formulation

$x_{ij}=1$, if processor P_i is the j-th to be activated and to receive data
$x_{ij}=0$, otherwise

$\alpha_{ij}>0$, is the amount of data sent to P_i if it is the j-th to be activated
$\alpha_{ij}=0$, otherwise

t_j is the time in which the j-th processor to be activated starts receiving its data

$$T^* = \text{minimum } T \tag{1}$$

subject to:

$$\sum_{i=1}^{n} x_{ij} \leq 1 \qquad j = 1, \ldots, n \tag{2}$$

$$\sum_{j=1}^{n} x_{ij} \leq 1 \qquad i = 1, \ldots, n \tag{3}$$

$$\sum_{i=1}^{n} x_{ij} \geq \sum_{i=1}^{n} x_{i,j+1} \qquad j = 1, \ldots, n-1 \tag{4}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{ij} = W \tag{5}$$

$$\alpha_{ij} \leq W x_{ij}$$

$$t_1 = 0$$

First processor is activated at time $t_1=0$

$$t_j \geq t_{j-1} + \sum_{i=1}^{n} (g_i x_{i,j-1} + G_i \alpha_{i,j-1}) \qquad j = 2, \ldots, n \tag{8}$$

$$t_j + \sum_{i=1}^{n} (g_i x_{ij} + (G_i + w_i)\alpha_{ij}) = T \qquad j = 1, \ldots, n \tag{9}$$

$$x_{ij} \in \{0, 1\} \qquad i, j = 1, \ldots, n \tag{10}$$

$$\alpha_{ij} \geq 0 \qquad i, j = 1, \ldots, n. \tag{11}$$

Formulation

$x_{ij}=1$, if processor P_i is the j-th to be activated and to receive data

$x_{ij}=0$, otherwise

$\alpha_{ij}>0$, is the amount of data sent to P_i if it is the j-th to be activated

$\alpha_{ij}=0$, otherwise

t_j is the time in which the j-th processor to be activated starts receiving its data

$$T^* = \text{minimum } T \tag{1}$$

subject to:

$$\sum_{i=1}^{n} x_{ij} \leq 1 \qquad j=1,\ldots,n \tag{2}$$

$$\sum_{j=1}^{n} x_{ij} \leq 1 \qquad i=1,\ldots,n \tag{3}$$

$$\sum_{i=1}^{n} x_{ij} \geq \sum_{i=1}^{n} x_{i,j+1}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_{ij} = W$$

$$\alpha_{ij} \leq W x_{ij}$$

$$t_1 = 0$$

$$t_j \geq t_{j-1} + \sum_{i=1}^{n} (g_i x_{i,j-1} + G_i \alpha_{i,j-1}) \qquad j=2,\ldots,n \tag{8}$$

$$t_j + \sum_{i=1}^{n} (g_i x_{ij} + (G_i + w_i)\alpha_{ij}) = T \qquad j=1,\ldots,n \tag{9}$$

$$x_{ij} \in \{0,1\} \qquad i,j=1,\ldots,n \tag{10}$$

$$\alpha_{ij} \geq 0 \qquad i,j=1,\ldots,n. \tag{11}$$

Communication link is sequentially used: j-th activated processor starts receiving data after (j-1)-th finishes

Formulation

x_{ij}=1, if processor P_i is the j-th to be activated and to receive data
x_{ij}=0, otherwise

α_{ij}>0, is the amount of data sent to P_i if it is the j-th to be activated
α_{ij}=0, otherwise

t_j is the time in which the j-th processor to be activated starts receiving its data

$$T^* = \text{minimum } T \tag{1}$$

subject to:

$$\sum_{i=1}^{n} x_{ij} \leq 1 \qquad j = 1, \ldots, n \tag{2}$$

$$\sum_{j=1}^{n} x_{ij} \leq 1 \qquad i = 1, \ldots, n \tag{3}$$

$$\sum_{i=1}^{n} x_{ij} \geq \sum_{i=1}^{n} x_{i,j+1} \qquad j = 1, \ldots, n-1 \tag{4}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{ij} = W \tag{5}$$

$$\alpha_{ij} \leq W x_{ij} \qquad i, j = 1, \ldots, n \tag{6}$$

$$t_1 = 0 \tag{7}$$

$$t_j \geq t_{j-1} + \sum_{i=1}^{n} (g_i x_{i,j-1} + G_i \alpha_{i,j-1}) \qquad j = 2, \ldots, n \tag{8}$$

$$t_j + \sum_{i=1}^{n} (g_i x_{ij} + (G_i + w_i)\alpha_{ij}) = T \qquad j = 1, \ldots, n \tag{9}$$

$$x_{ij} \in \{0, 1\}$$

$$\alpha_{ij} \geq 0$$

All processors finish at the same time T

Linear-time algorithm for
a given activation order

Linear-time algorithm for a given activation order

- Blazewicz and Drozdowski (1997): if the activation order and the number of processors are known, the optimal loads are:

$$\alpha_k = \alpha_\ell \prod_{j=k+1}^{\ell} f_j + \sum_{j=k+1}^{\ell} \left(\frac{g_j}{w_{j-1}} \prod_{i=k+1}^{j-1} f_i\right), \quad k = 1, \ldots, \ell-1 \tag{14}$$

$$\alpha_\ell = \frac{W - \sum_{k=1}^{\ell-1} \sum_{j=k+1}^{\ell} \left(\frac{g_j}{w_{j-1}} \prod_{i=k+1}^{j-1} f_i\right)}{1 + \sum_{k=1}^{\ell-1} \prod_{j=k+1}^{\ell} f_j} \tag{15}$$

$$\boxed{V(\ell) = \frac{g_\ell}{w_{\ell-1}} \sum_{k=1}^{\ell-1} \prod_{i=k+1}^{\ell-1} f_i + V(\ell-1).}$$

Linear-time algorithm for a given activation order

- $F(\ell) = \sum_{k=1}^{\ell-1} \prod_{i=k+1}^{\ell-1} f_i$ may be recursively defined as

$$F(1) = 0, F(2) = 1, \text{ and } F(\ell) = 1 + F(\ell - 1) f_{\ell-1}$$

- $$V(\ell) \quad = \quad \frac{g_\ell}{w_{\ell-1}} F(\ell) + V(\ell - 1)$$ may be computed in time O(1)

- Optimal solution has the maximum number ℓ^* of processors such that

$$V(\ell^*) \leq W$$

Linear-time algorithm for a given activation order

- Algorithm:
  - Compute F(k) for k=1,…, n in time O(n)
  - Compute V(k) for k=1,…, n in time O(n)
  - Optimal number of processors is the largest number of processors k such that V(k) ≤ W
  - Load assigned to each processor can be computed in time O(n) as described by Blazewicz and Drozdowski (1997)

Fast constructive heuristic with feedback

Constructive feedback heuristic

- Heuristic for scheduling divisible loads may be seen as any algorithm that generates a “good” activation order and computes the associated optimal loads.
- Constructive feedback heuristic makes use of the idea of equivalent processors
- Each solution is uniquely associated with:
  - activation order given by a vector π
  - makespan T

Constructive feedback heuristic

- Equivalent processor:
  - Given a time period T, if a load $\alpha_i = (T-g_i) / (w_i+G_i)$ is sent to P_i then it remains busy with communication and processing for this full time period
  - Equivalent to a processor P_i^{eq} with the same processing power, no communication latency, and throughput $1/G_i^{eq} = 1/[G_i + (g_i / \alpha_i)]$
- Optimal activation order for a system with no latencies: processors with higher communication throughput receive data first

Constructive feedback heuristic

Create activation order π with higher throughput processors first

UB = optimal makespan for activation order π

Repeat

BestOrder = π

T^* = UB

Compute new order π

UB = optimum makespan for new activation order π

Until UB $\geq$ T^*

Constructive feedback heuristic

Create activation order π with higher throughput processors first

UB = optimal makespan for activation order π

Repeat

BestOrder = π

T^* = UB

For j = 1, ..., n do

Compute equivalent processor P_i^{eq} for each P_i not in $\pi[1], ..., \pi[j-1]$

$\pi[j]$ = processor whose equivalent has the highest throughput

Update remaining time UB by subtracting the time taken by that processor

UB = optimum makespan for activation order π

Until UB ≥ T^*

Computational experiments

Computational experiments

- 120 grid configurations
  - Number of processors: 10, 20, 40, 80, and 160
  - 24 configurations of w_i, G_i, and g_i
- Load W: 100, 200, 400, 800, 1600, and 3200
- CPLEX time limit 3600 seconds

Computational experiments

CPLEX solved 490 out of 720 test instances

TABLE I

OPTIMAL SOLUTIONS AND RUNNING TIMES IN SECONDS

			$n = 10$		$n = 20$		$n = 40$		$n = 80$		$n = 160$	
w_i	g_i	G_i	opt.	time	opt.	time	opt.	time	opt.	time	opt.	time
low	low	low	18	0.25	18	0.81	18	43.81	5	–	1	–
low	low	high	18	0.05	18	0.08	18	0.36	18	1.37	18	4.94
low	high	low	18	0.08	18	0.20	18	0.46	18	2.78	18	9.27
low	high	high	18	0.14	18	0.25	18	0.50	18	1.20	18	3.57
high	low	low	18	14.64	0	–	0	–	0	–	0	–
high	low	high	18	0.06	18	9.37	16	–	0	–	0	–
high	high	low	18	2.85	5	–	1	–	2	–	0	–
high	high	high	18	0.13	18	0.68	18	52.52	8	–	2	–

Computational experiments

Feedback heuristic found optimal solutions for 398 out of the 490 instances for which CPLEX found the optimum

TABLE II

PROVABLE OPTIMAL SOLUTIONS FOUND BY THE FEEDBACK HEURISTIC

			Processors (n)				
w_i	g_i	G_i	10	20	40	80	160
low	low	low	11	10	4	1	0
low	low	high	18	18	18	18	17
low	high	low	13	17	17	11	13
low	high	high	18	18	18	18	14
high	low	low	18	–	–	–	–
high	low	high	18	18	16	–	–
high	high	low	16	1	0	1	–
high	high	high	14	14	5	3	2

Computational experiments

Average deviation from optimal value smaller than 0.5% for most of the remaining 92 instances solved to optimality by CPLEX

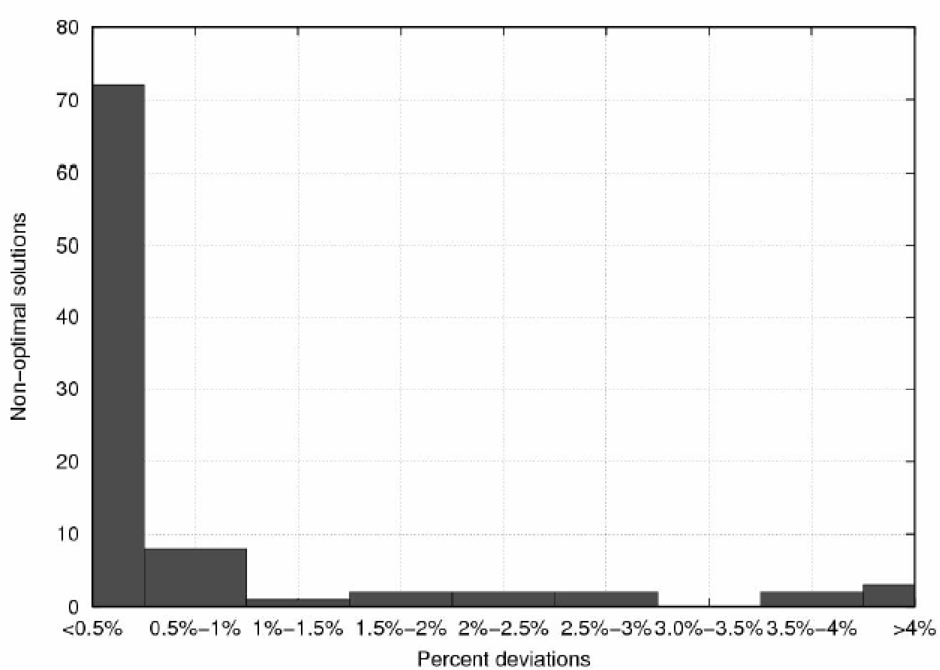


Fig. 5. Percent deviation of non-optimal makespans

Heuristic ran for 3ms on average and never for more than 32ms

Multiple-installment mixed integer programming formulation

Formulation

In this case, model also determines the optimal number of installments.

Model 2 Multiple-installment mixed integer program

$$T^* = \text{minimum } T \qquad (18)$$

subject to:

$$\sum_{i=1}^{n} x_{kij} \leq 1 \qquad j = 1, \ldots, n, \; k = 1, \ldots, p \qquad (19)$$

$$\sum_{j=1}^{n} x_{kij} \leq 1 \qquad i = 1, \ldots, n, \; k = 1, \ldots, p \qquad (20)$$

$$\sum_{i=1}^{n} x_{kij} \geq \sum_{i=1}^{n} x_{ki,j+1} \qquad j = 1, \ldots, n-1, \; k = 1, \ldots, p \qquad (21)$$

$$\sum_{k=1}^{p} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{kij} = W \qquad (22)$$

$$\alpha_{kij} \leq W x_{kij} \qquad i, j = 1, \ldots, n, \; k = 1, \ldots, p \qquad (23)$$

$$t_{11} = 0 \qquad (24)$$

$$t_{kj} \geq t_{k,j-1} + \sum_{i=1}^{n} (g_i x_{ki,j-1} + G_i \alpha_{ki,j-1}) \qquad j = 2, \ldots, n, \; k = 1, \ldots, p \qquad (25)$$

$$t_{k,j} + \sum_{i=1}^{n} (g_i x_{kij} + G_i \alpha_{kij}) \geq t_{k-1,j} + \sum_{i=1}^{n} (g_i x_{k-1,ij} + (G_i + w_i) \alpha_{k-1,ij}) \qquad j = 1, \ldots, n, \; k =, \ldots, p \qquad (26)$$

$$t_{k1} \geq t_{k-1,n} + \sum_{i=1}^{n} (g_i x_{k-1,in} + G_i \alpha_{k-1,in}) \qquad k = 2, \ldots, p \qquad (27)$$

$$t_{pj} + \sum_{i=1}^{n} (g_i x_{pij} + (G_i + w_i) \alpha_{pij}) = T \qquad j = 1, \ldots, n \qquad (28)$$

$$x_{kij} \in \{0, 1\} \qquad i, j = 1, \ldots, n, \; k = 1, \ldots, p \qquad (29)$$

$$\alpha_{kij} \geq 0 \qquad i, j = 1, \ldots, n, \; k = 1, \ldots, p. \qquad (30)$$

Formulation

Preliminary results show significant improvements in the makespans are possible, with respect to those obtained by single- and multi-round heuristics.

Model 2 Multiple-installment mixed integer

$T^* = \text{minimum } T$

subject to:

$\sum_{i=1}^{n} x_{kij} \le 1 \qquad j = 1,\ldots,n, \quad k = 1,\ldots,p \quad (19)$

$\sum_{j=1}^{n} x_{kij} \le 1 \qquad i = 1,\ldots,n, \quad k = 1,\ldots,p \quad (20)$

$\sum_{i=1}^{n} x_{kij} \ge \sum_{i=1}^{n} x_{ki,j+1} \qquad j = 1,\ldots,n-1, \quad k = 1,\ldots,p \quad (21)$

$\sum_{k=1}^{p} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{kij} = W \quad (22)$

$\alpha_{kij} \le W x_{kij} \qquad i,j = 1,\ldots,n, \quad k = 1,\ldots,p \quad (23)$

$t_{11} = 0 \quad (24)$

$t_{kj} \ge t_{k,j-1} + \sum_{i=1}^{n} (g_i x_{ki,j-1} + G_i \alpha_{ki,j-1}) \qquad j = 2,\ldots,n, \quad k = 1,\ldots,p \quad (25)$

$t_{k,j} + \sum_{i=1}^{n} (g_i x_{kij} + G_i \alpha_{kij}) \ge t_{k-1,j} + \sum_{i=1}^{n} (g_i x_{k-1,ij} + (G_i + w_i)\alpha_{k-1,ij}) \qquad j = 1,\ldots,n, \quad k =,\ldots,p \quad (26)$

$t_{k1} \ge t_{k-1,n} + \sum_{i=1}^{n} (g_i x_{k-1,in} + G_i \alpha_{k-1,in}) \qquad k = 2,\ldots,p \quad (27)$

$t_{pj} + \sum_{i=1}^{n} (g_i x_{pij} + (G_i + w_i)\alpha_{pij}) = T \qquad j = 1,\ldots,n \quad (28)$

$x_{kij} \in \{0,1\} \qquad i,j = 1,\ldots,n, \quad k = 1,\ldots,p \quad (29)$

$\alpha_{kij} \ge 0 \qquad i,j = 1,\ldots,n, \quad k = 1,\ldots,p. \quad (30)$

Concluding remarks

- New mixed integer programming formulations for single- and multi-round schedulings.
- Linear-time algorithm for the special case in which the processor activation order is known.
- Fast and effective greedy-with-feedback heuristic.
- Randomized multistart version of feedback heuristic with local search.
- Extension to multi-round schedulings.