Mapping filter services on heterogeneous platforms

Anne Benoit, Fanny Dufossé, Yves Robert

ENS Lyon

June 2, 2009

Introduction

The problem:

- processing of a data flow
- filter services with different selectivities and costs
- precedence constraints between services
- servers with different speeds
- one-to-one mappings

The objective:

- minimizing period
- minimizing latency
- bi-criteria: minimize latency for a fixed period

The model

Example: In a list L of numbers:

- a first filter removes odd numbers ($\sigma_1 = 1/2$)
- a second filter transmits only multiples of 3 ($\sigma_1 = 1/3$)

The resulting list contains all multiples of 6 in L.

The model

Example: In a list L of numbers:

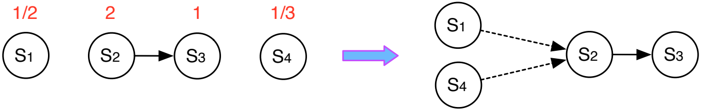
- a first filter removes odd numbers ($\sigma_1 = 1/2$)
- a second filter transmits only multiples of 3 ($\sigma_1 = 1/3$)

The resulting list contains all multiples of 6 in L.
Hypotheses:

- No communication cost
- Join operation of null cost
- Selectivities are independent

Playing with selectivities

- Service S_i transforms (filters) data of size δ to size $\sigma_i \times \delta$
- Computation cost depends on the data size (previous σ)
- May add dependencies to exploit selectivity



- S_1 and S_4 process file of initial size 1; S_1 removes even numbers and S_2 removes not multiples of 3.
- Combined file of size $\frac{1}{2}.\frac{1}{3} = \frac{1}{6}$ (no cost for join)
- S_2 duplicates the file
- S_3 processes but does not alter the file

References

M. P. Alessandro Agnetis, Paolo Detti and M. S. Sodhi.
Sequencing unreliable jobs on parallel machines.
Journal on Scheduling, May 2008.

U. Srivastava, K. Munagala, and J. Burge.
Ordering pipelined query operators with precedence constraints.
Technical report, Stanford University, November 2005.

U. Srivastava, K. Munagala, and J. Widom.
Operator placement for in-network stream query processing.
In *PODS '05: Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 250–258, New York, NY, USA, 2005. ACM.

U. Srivastava, K. Munagala, J. Widom, and R. Motwani.
Query optimization over web services.
In *VLDB '06: Proceedings of the 32nd international conference on Very large data bases*, pages 355–366. VLDB Endowment, 2006.

The instances

The problems depend on:

- the criteria: MinPeriod, MinLatency or BiCriteria
- the platform: Hom or Het
- the dependence constraints: NoPrec or Prec

The instances

The problems depend on:

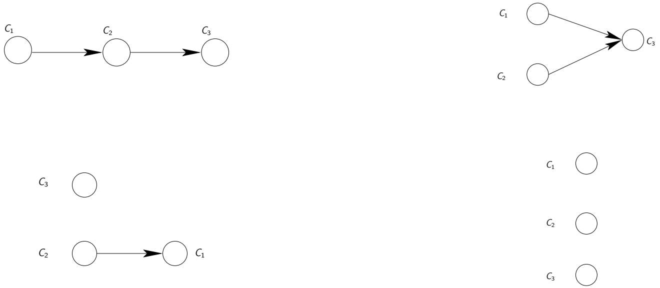
- the criteria: MinPeriod, MinLatency or BiCriteria
- the platform: Hom or Het
- the dependence constraints: NoPrec or Prec

The instances: $\mathcal{A} = (\mathcal{F}, \mathcal{G}, \mathcal{S})$ with:

- The services: $\mathcal{F} = \{C_1, C_2, \dots, C_n\}$
- The precedence constraints: $\mathcal{G} \subset \mathcal{F} \times \mathcal{F}$
- The servers: $\mathcal{S} = \{S_1, S_2, \dots, S_p\}$

The problem

Example for 3 independent services:
The plan?



The mapping?

$$(C_1, S_2), (C_2, S_1), (C_3, S_3)$$

$$(C_1, S_3), (C_2, S_2), (C_3, S_1)$$

Example

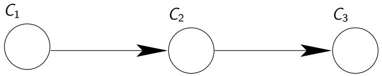


Figure: Chaining services.

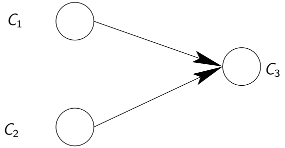


Figure: Combining selectivities

$$\mathcal{P} = \max\left(\frac{c_1}{s_1}, \frac{\sigma_1 c_2}{s_2}, \frac{\sigma_1\sigma_2 c_3}{s_3}\right) \qquad \mathcal{P} = \max\left(\frac{c_1}{s_1}, \frac{c_2}{s_2}, \frac{\sigma_1\sigma_2 c_3}{s_3}\right)$$

$$\mathcal{L} = \frac{c_1}{s_1} + \frac{\sigma_1 c_2}{s_2} + \frac{\sigma_1\sigma_2 c_3}{s_3} \qquad \mathcal{L} = \max\left(\frac{c_1}{s_1}, \frac{c_2}{s_2}\right) + \frac{\sigma_1\sigma_2 c_3}{s_3}$$

Example

- $c_1 = 1,\ c_2 = 4,\ c_3 = 10$
- $\sigma_1 = \frac{1}{2},\ \sigma_2 = \sigma_3 = \frac{1}{3}$
- $s_1 = 1,\ s_2 = 2$ and $s_3 = 3$

Example

- $c_1 = 1,\ c_2 = 4,\ c_3 = 10$
- $\sigma_1 = \frac{1}{2},\ \sigma_2 = \sigma_3 = \frac{1}{3}$
- $s_1 = 1,\ s_2 = 2$ and $s_3 = 3$

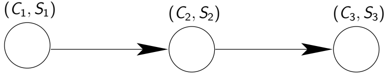


Figure: Optimal plan for period.

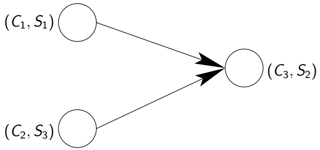


Figure: Optimal plan for latency

$$P = 1$$
$$L = \frac{5}{2}$$

$$L = \frac{13}{6}$$
$$P = \frac{4}{3}$$

General structure of optimal solutions

The instance : $C_1, ..., C_n, S_1, ..., S_n$ with

- $\sigma_1, ..., \sigma_p \leq 1$
- $\sigma_{p+1}, ..., \sigma_n \geq 1$

General structure of optimal solutions

The instance : $C_1, ..., C_n, S_1, ..., S_n$ with

- $\sigma_1, ..., \sigma_p \leq 1$
- $\sigma_{p+1}, ..., \sigma_n \geq 1$

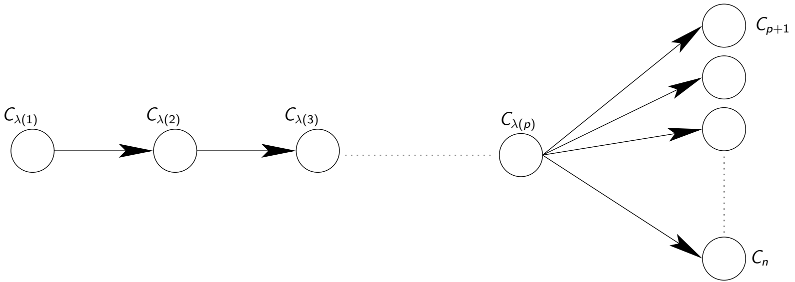


Figure: General structure

Homogeneous case without precedence constraints

The instance : $C_1, ..., C_n$ with

- $c_1 \leq c_2 \leq ... \leq c_p$
- $\sigma_1, ..., \sigma_p < 1$
- $\sigma_{p+1}, ..., \sigma_n \geq 1$

The matching: $C_1 \rightarrow C_2 \rightarrow ... \rightarrow C_p$

Homogeneous case with precedence constraints

Computing the optimal subgraph for C in the graph G is polynomial. Paper [Srivastava et al] presents a polynomial time algorithm using a min-cut algorithm.

Homogeneous case with precedence constraints

Computing the optimal subgraph for C in the graph G is polynomial.
Paper [Srivastava et al] presents a polynomial time algorithm using a min-cut algorithm.
Optimal algorithm for period in homogeneous case:
We add the nodes step by step.
At each step, we place the available service with minimal possible period.

Proof: NP-completeness of MinPeriod-Het

Problem (RN3DM)

Given an integer vector $A = (A[1], \ldots, A[n])$ of size n, does there exist two permutations λ_1 and λ_2 of $\{1, 2, \ldots, n\}$ such that

$$\forall 1 \leq i \leq n, \quad \lambda_1(i) + \lambda_2(i) = A[i]$$

Proof: NP-completeness of MinPeriod-Het

Problem (RN3DM)

Given an integer vector $A = (A[1], \dots, A[n])$ of size n, does there exist two permutations λ_1 and λ_2 of $\{1, 2, \dots, n\}$ such that

$$\forall 1 \leq i \leq n, \quad \lambda_1(i) + \lambda_2(i) = A[i]$$

The associated instance :

- $c_i = 2^{A[i]}$
- $\sigma_i = 1/2$
- $s_i = 2^i$
- $P = 2$

Proof: NP-completeness of MinPeriod-Het

Problem (RN3DM)

Given an integer vector $A = (A[1], \ldots, A[n])$ of size n, does there exist two permutations λ_1 and λ_2 of $\{1, 2, \ldots, n\}$ such that

$$\forall 1 \leq i \leq n, \quad \lambda_1(i) + \lambda_2(i) = A[i]$$

The associated instance :

- $c_i = 2^{A[i]}$
- $\sigma_i = 1/2$
- $s_i = 2^i$
- $P = 2$

$$\begin{aligned} & \forall 1 \leq i \leq n, \quad \lambda_1(i) + \lambda_2(i) \geq A[i] \\ \iff \; & \forall 1 \leq i \leq n, \quad \left(\frac{1}{2}\right)^{\lambda_1(i)-1} \times \frac{2^{A[i]}}{2^{\lambda_2(i)}} \leq 2 \end{aligned}$$

Inapproximability of MinPeriod-Het

Proposition

For any $K > 0$, there exists no K-approximation algorithm for MinPeriod-NoPrec-Het, *unless P=NP.*

Inapproximability of MinPeriod-Het

Proposition

For any $K > 0$, there exists no K-approximation algorithm for MinPeriod-NoPrec-Het, *unless P=NP.*

Reduction from RN3DM:

- $c_i = K^{A[i]-1}$
- $\sigma_i = 1/K$
- $s_i = K^i$
- $P = 1$

Integer linear program

The variables:

- $t_{i,u} = 1$ if service C_i is assigned to server S_u
- $s_{i,j} = 1$ if service C_i is an ancestor of C_j
- M is the logarithm of the optimal period

Integer linear program

The variables:

- $t_{i,u} = 1$ if service C_i is assigned to server S_u
- $s_{i,j} = 1$ if service C_i is an ancestor of C_j
- M is the logarithm of the optimal period

The constraints:

- $\forall i, \quad \sum_u t_{i,u} = 1$
- $\forall u, \quad \sum_i t_{i,u} = 1$
- $\forall i, j, k, \quad s_{i,j} + s_{j,k} - 1 \leq s_{i,k}$
- $\forall i, s_{i,i} = 0$
- $\forall i, \quad \log c_i - \sum_u t_{i,u} \log s_u + \sum_k s_{k,i} \log \sigma_k \leq M$

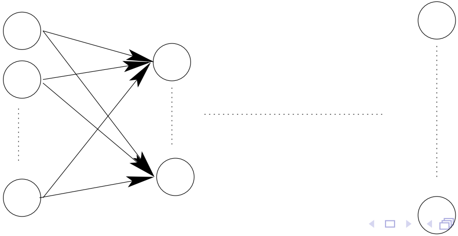
The objective function: Minimize M

Structure of the optimal plan

Proposition

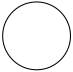
Let $C_1, \dots, C_n, S_1, \dots, S_n$ be an instance of MinLatency. *Then, the optimal latency is obtained with a plan G such that, for any $v_1 = (C_{i_1}, S_{u_1})$, $v_2 = (C_{i_2}, S_{u_2})$,*

1. *If $d_{i_1}(G) = d_{i_2}(G)$, they have the same predecessors and the same successors in G.*
2. *If $d_{i_1}(G) > d_{i_2}(G)$ and $\sigma_{i_2} \leq 1$, then $c_{i_1}/s_{u_1} < c_{i_2}/s_{u_2}$.*
3. *All nodes with a service of selectivity $\sigma_i > 1$ are leaves ($d_i(G) = 0$).*



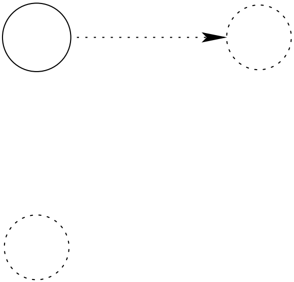
Principle of the algorithm

C_1

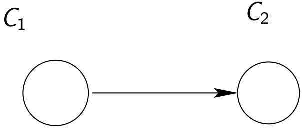


Principle of the algorithm

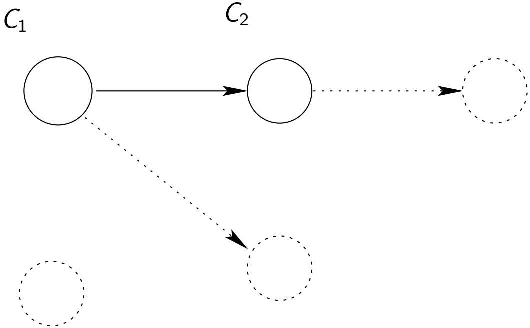
C_1



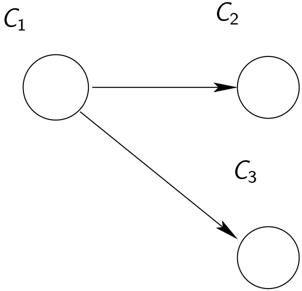
Principle of the algorithm



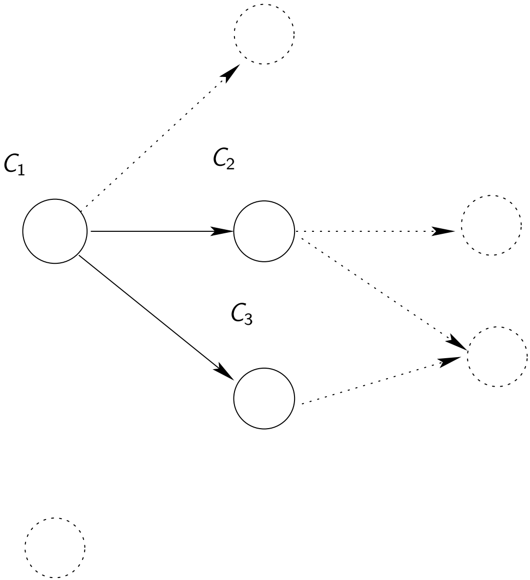
Principle of the algorithm



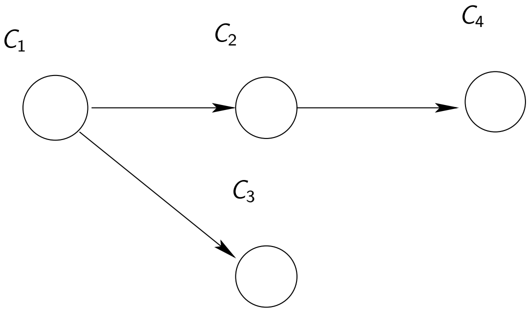
Principle of the algorithm



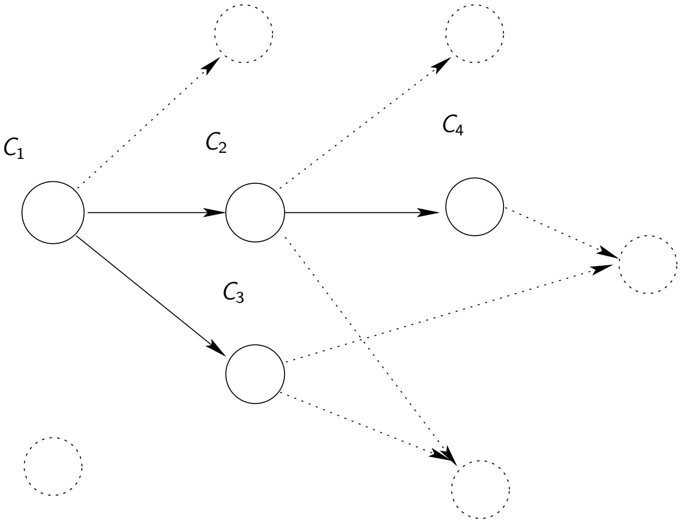
Principle of the algorithm



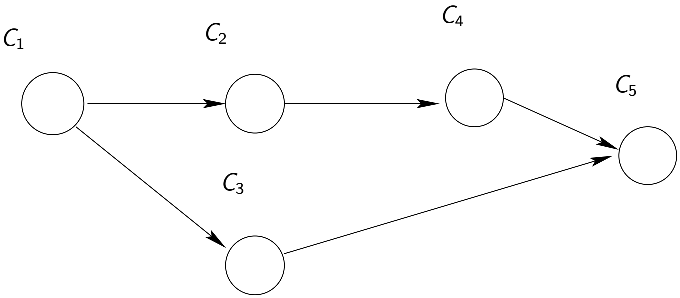
Principle of the algorithm



Principle of the algorithm



Principle of the algorithm



Algorithm without dependence constraints

Data: n services of costs $c_1 \leq \cdots \leq c_n$ and of selectivities $\sigma_1, ..., \sigma_n \leq 1$
Result: a plan G optimizing the latency
G is the graph reduced to node C_1;
for $i = 2$ **to** n **do**
    for $j = 0$ **to** $i - 1$ **do**
        Compute the completion time t_j of C_i in G with predecessors $C_1, ..., C_j$;
    end
    Choose j such that $t_j = \min_k\{t_k\}$;
    Add the node C_i and the edges $C_1 \rightarrow C_i, \ldots, C_j \rightarrow C_i$ to G;
end

Algorithm 1: Optimal algorithm for MinLatency-NoPrec-Hom.

G is the graph reduced to the node C of minimal cost with no predecessor in $\mathcal{G}$;

for $i = 2$ **to** n **do**

Let S be the set of services not yet in G and such that their set of predecessors in $\mathcal{G}$ is included in G;

for $C \in S$ **do**

for $C' \in G$ **do**

Compute the set S' minimizing the product of selectivities among services of latency less than $L_G(C')$, and including all predecessors of C in $\mathcal{G}$;

end

Let S_C be the set that minimizes the latency of C in G and L_C be this latency;

end

Choose a service C such that $L_C = \min\{L_{C'}, C' \in S\}$;

Add to G the node C, and $\forall C' \in S_C$, the edge $C' \rightarrow C$;

end

Algorithm 2: Optimal algorithm for MINLATENCY-PREC-HOM.

1. Framework
2. Period
   - General structure of optimal solutions
   - Case of homogeneous servers
   - NP-completeness of MinPeriod-Het
   - Integer linear program
3. Latency
   - General structure of optimal solutions
   - Polynomial algorithm on homogeneous platforms
   - NP-completeness of problem MinLatency-NoPrec-Het
   - Integer linear program
4. Bi-criteria problem
5. Conclusion

Proof:NP-completeness of MinLatency-Het

Lemma

Let $C_1, ..., C_n, S_1, ..., S_n$ be an instance such that $\forall i$, c_i and s_i are integer power of 2 and $\sigma_i \leq \frac{1}{2}$. Then the optimal latency is obtained with a plan G such that

1. *Proposition 2 is verified;*
2. *for all nodes (C_{i_1}, S_{u_1}) and (C_{i_2}, S_{u_2}) with $d_{i_1}(G) = d_{i_2}(G)$, we have $\frac{c_{i_1}}{s_{u_1}} = \frac{c_{i_2}}{s_{u_2}}$.*

Proof:NP-completeness of MinLatency-Het

Lemma

Let $C_1, ..., C_n, S_1, ..., S_n$ be an instance such that $\forall i$, c_i and s_i are integer power of 2 and $\sigma_i \leq \frac{1}{2}$. Then the optimal latency is obtained with a plan G such that

1. *Proposition 2 is verified;*
2. *for all nodes (C_{i_1}, S_{u_1}) and (C_{i_2}, S_{u_2}) with $d_{i_1}(G) = d_{i_2}(G)$, we have $\frac{c_{i_1}}{s_{u_1}} = \frac{c_{i_2}}{s_{u_2}}$.*

- $c_i = 2^{A[i] \times n + (i-1)}$
- $\sigma_i = \left(\frac{1}{2}\right)^n$
- $s_i = 2^{n \times (i+1)}$
- $L = 2^n - 1$

Inapproximability of MinLatency-Het

Proposition

For any $K > 0$, there exists no K-approximation algorithm for MinLatency-NoPrec-Het, *unless P=NP.*

Inapproximability of MinLatency-Het

Proposition

For any $K > 0$, there exists no K-approximation algorithm for MinLatency-NoPrec-Het, *unless P=NP.*

Reduction from RN3DM

- $c_i = K^{A[i] \times n + (i-1)}$
- $\sigma_i = \left(\frac{1}{K}\right)^n$
- $s_i = K^{n \times (i+1)}$
- $L = \frac{K^n - 1}{K - 1}$

The variables:

- $z(i, u, e) = 1$ if the service C_i is associated to the server S_u and its set of predecessors is $e \subset \mathcal{C}$.
- $t(i)$ is the completion time of C_i
- M is the optimal latency

The variables:

- $z(i, u, e) = 1$ if the service C_i is associated to the server S_u and its set of predecessors is $e \subset \mathcal{C}$.
- $t(i)$ is the completion time of C_i
- M is the optimal latency

The constraints:

- $\forall u \in \mathcal{S}, \quad \sum_{i \in \mathcal{C}} \sum_{e \subset \mathcal{C}} z(i, u, e) = 1$
- $\forall i \in \mathcal{C}, \quad \sum_{u \in \mathcal{S}} \sum_{e \subset \mathcal{C}} z(i, u, e) = 1$
- $\forall i, i' \in \mathcal{C}, \forall u, u' \in \mathcal{S}, \forall e, e' \subset \mathcal{C}, e \not\subseteq e', i \in e', \quad z(i, u, e) + z(i', u', e') \leq 1$
- $\forall u \in \mathcal{S}, \forall e \subset \mathcal{C}, \forall i \in e, \ z(i, u, e) = 0$
- $\forall i \in \mathcal{C}, \forall e \subset \mathcal{C}, \forall k \in e, \quad t(i) \geq \sum_{u \in \mathcal{S}} z(i, u, e) \left(\frac{c_i}{s_u} * \prod_{C_j \in e} \sigma_j + t(k) \right)$
- $\forall i \in \mathcal{C}, \quad t(i) \geq \sum_u z(i, u, e) \frac{c_i}{s_u} * \prod_{C_j \in e} \sigma_j$
- $\forall i \in \mathcal{C}, \quad t(i) \leq M$

The objective function: Minimize M

Data: n services of costs $c_1 \leq \cdots \leq c_n$ and of selectivities $\sigma_1, ..., \sigma_n \leq 1$ and a maximum throughput K

Result: a plan G optimizing the latency with a throughput less than K

G is the graph reduced to node C_1;

for $i = 2$ **to** n **do**

  for $j = 0$ **to** $i - 1$ **do**

    Compute the completion time t_j of C_i in G with predecessors $C_1, ..., C_j$;

  end

  Let $S = \{k | c_i \prod_{0 \leq k < i} \sigma_k \leq K\}$;

  Choose j such that $t_j = \min_{k \in S}\{t_k\}$;

  Add the node c_i and the edges $C_1 \rightarrow C_i, \ldots, C_j \rightarrow C_i$ to G;

end

Algorithm 3: Optimal algorithm for latency with a fixed throughput.

Conclusion

The results:

- MinLatency-Hom is polynomial
- MinPeriod-Het is NP-hard
- MinLatency-Het is NP-hard
- BiCriteria-Hom is polynomial

Future work:

- Model with communication costs (see SPAA'09)