Minimisation de la durée totale d'un projet multi-agent:
recherche du meilleur équilibre de Nash

CYRIL BRIAND[1,2], SANDRA U. NGUEVEU[1,2], PŘEMYSL ŠŮCHA[3]
{ngueveu@laas.fr, briand@laas.fr, suchap@fel.cvut.cz}

1) LAAS-CNRS

2) Université de Toulouse

3)

ANR. 08-BLAN-0331-02 "ROBOCOOP"

Introduction

- Multi-agent optimization
  - A set of agents involved in a common decision-making process
  - Every agent
    - controls its decision variables (*strategy*)
    - is interested in maximizing its profit
  - The agents' profit can depend on the strategies of the other agents

- Multi-objective optimization

$$\text{opt } F(x) = (f_1(x), \ldots, f_m(x)) \text{ s.t. } x \in \Omega$$

with

- $\mathcal{A} = \{A_1, \ldots, A_m\} =$ Agents set
- $x = (x_1, \ldots, x_m) =$ Vector of Agents' strategy
- $f_u(x) =$ Profit of A_u

Strategy requirements

- Efficiency
  - As every agent wants to maximize its profit
    - only non-dominated strategies are of interest
  - Multi-objective optimization: Pareto optima
- Stability
  - Given a strategy, no agent should be able to increase its profit by itself, while decreasing the profit of others
  - Game theory: Nash equilibria
- Price of cooperation
  - A global objective function (GOF)
  - Price of anarchy : PA=(GOF value in the worst NE)/(OPT)
  - Price of stability : PS=(GOF value in the best NE)/(OPT)

Outline

- Introduction
- Multi-Agent project scheduling (MAPS)
- Finding a minimum-makespan stable strategy
- Optimal reward sharing
- Conclusion

Problem statement

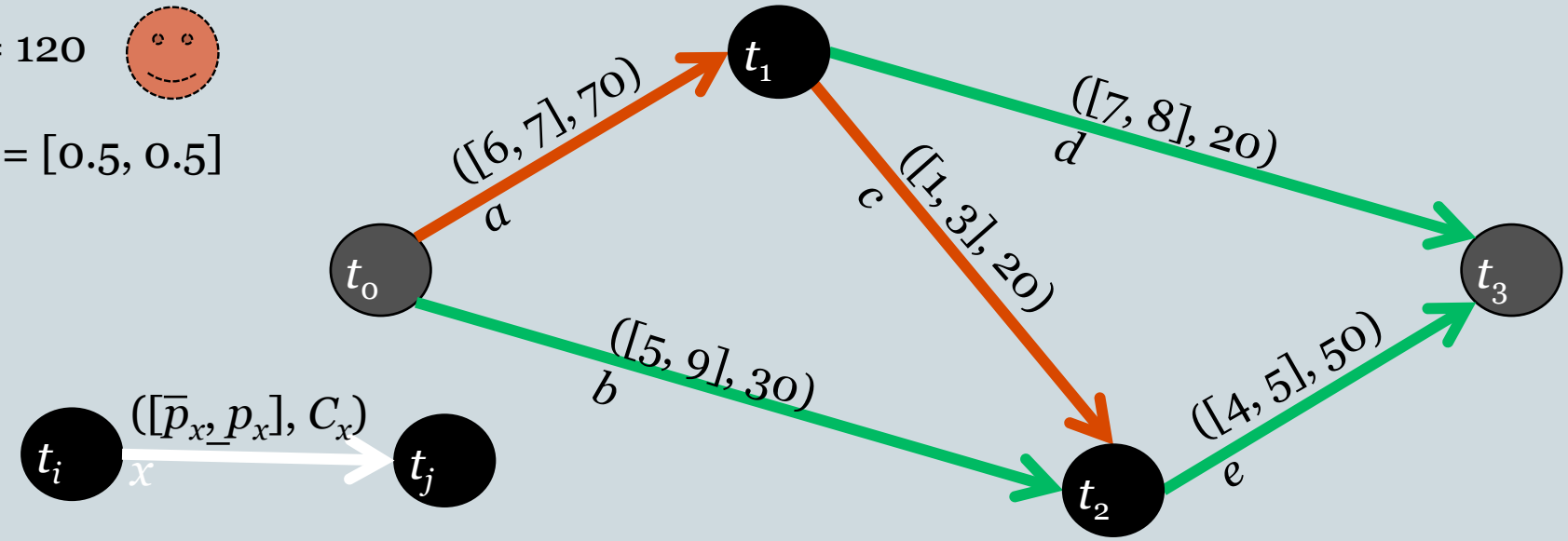
- A customer contracts a project
  - Activity-on-arc network $G(N,U)$, unlimited resource capacities
  - Precedence constraints $t_j - t_i - p_{i,j} \geq 0, \forall (i,j) \in U$
- Project activities are distributed among agents $\mathcal{A} = \{A_1, \ldots, A_m\}$
  - They control the durations of their activities and bear the compression costs
  - $c_{i,j}, p_{i,j} \in [\underline{p}_{i,j}, \overline{p}_{i,j}], (i,j) \in \mathcal{T}_u$
- The customer offers a daily reward π to favor makespan reduction
  - The total reward is shared among agents according to a sharing policy pre-established or to be-defined
  - $w = \{w_1, \ldots, w_m\}, \sum_{\mathcal{A}} w_u = 1$
- Agents want to maximize their profit
  - Agent's reward – Compression costs
  - $Z_u(S) = w_u \pi(\overline{D} - t_n) - \sum_{\{(i,j) \in \mathcal{T}_u\}} c_{i,j}(\overline{p}_{i,j} - p_{i,j})$

Example (inspired from Phillips and Dessouky, 1977)

- Activity-on-arc graph with 5 activities: {*a*,*b*,*c*,*d*,*e*}
- 2 agents: green (g) and red (r)
- Customer: π=120, w_g=1/2, w_r=1/2

π = 120

w_u = [0.5, 0.5]

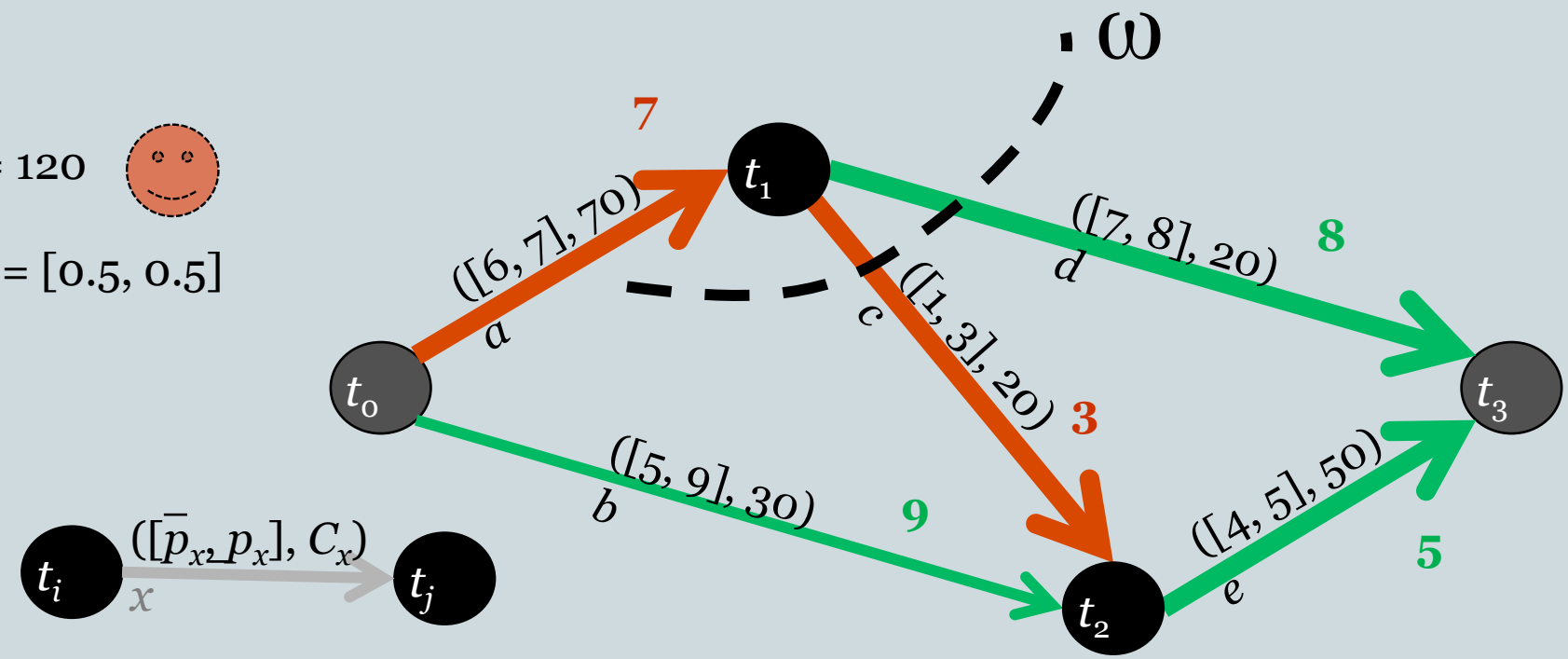


Example (inspired from Phillips and Dessouky, 1977)

t_3-t_0	Rewards	$Cost_g$	$Cost_r$	Z_g	Z_r	
15	0	0	0	0	0	S_0

$\pi = 120$

$w_u = [0.5, 0.5]$

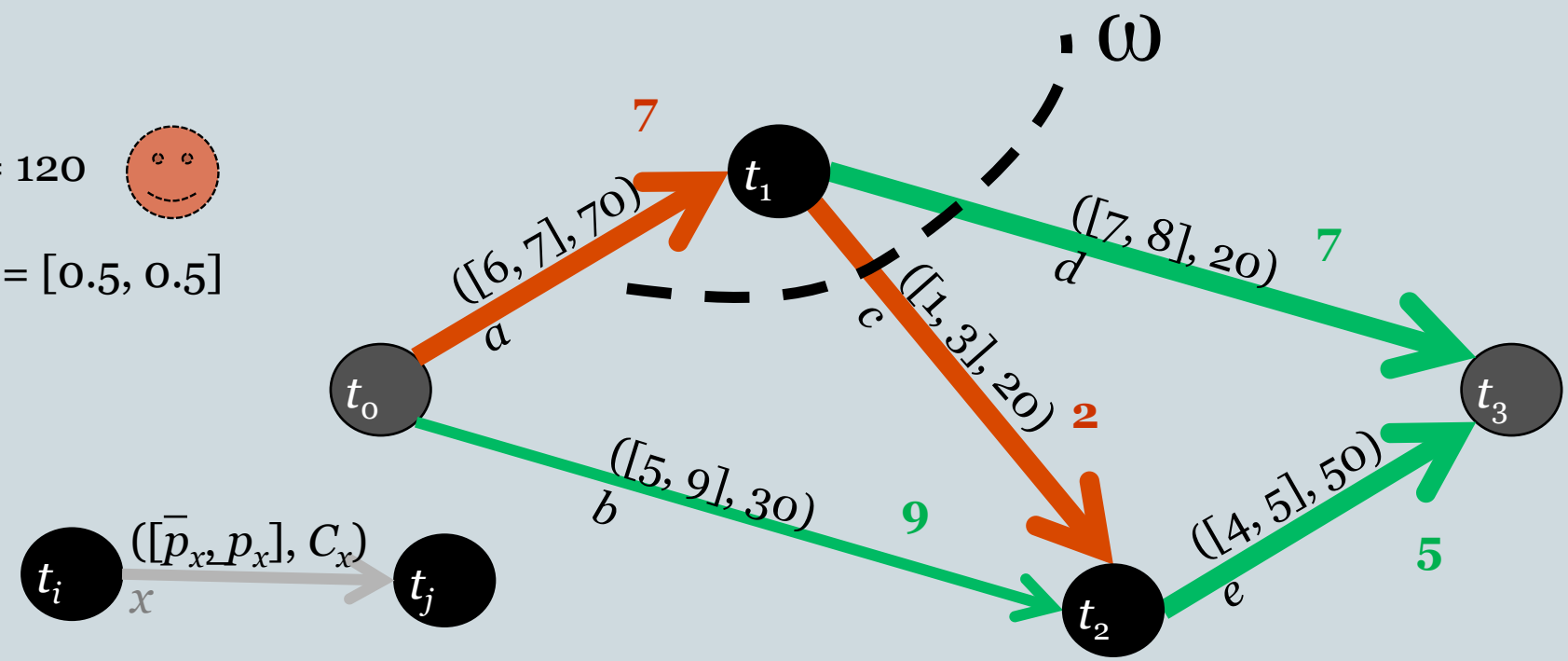


Example (inspired from Phillips and Dessouky, 1977)

t_3-t_0	Rewards	$Cost_g$	$Cost_r$	Z_g	Z_r	
15	0	0	0	0	0	S_0
14	120	20	20	40	40	S_1

$\pi = 120$

$w_u = [0.5, 0.5]$

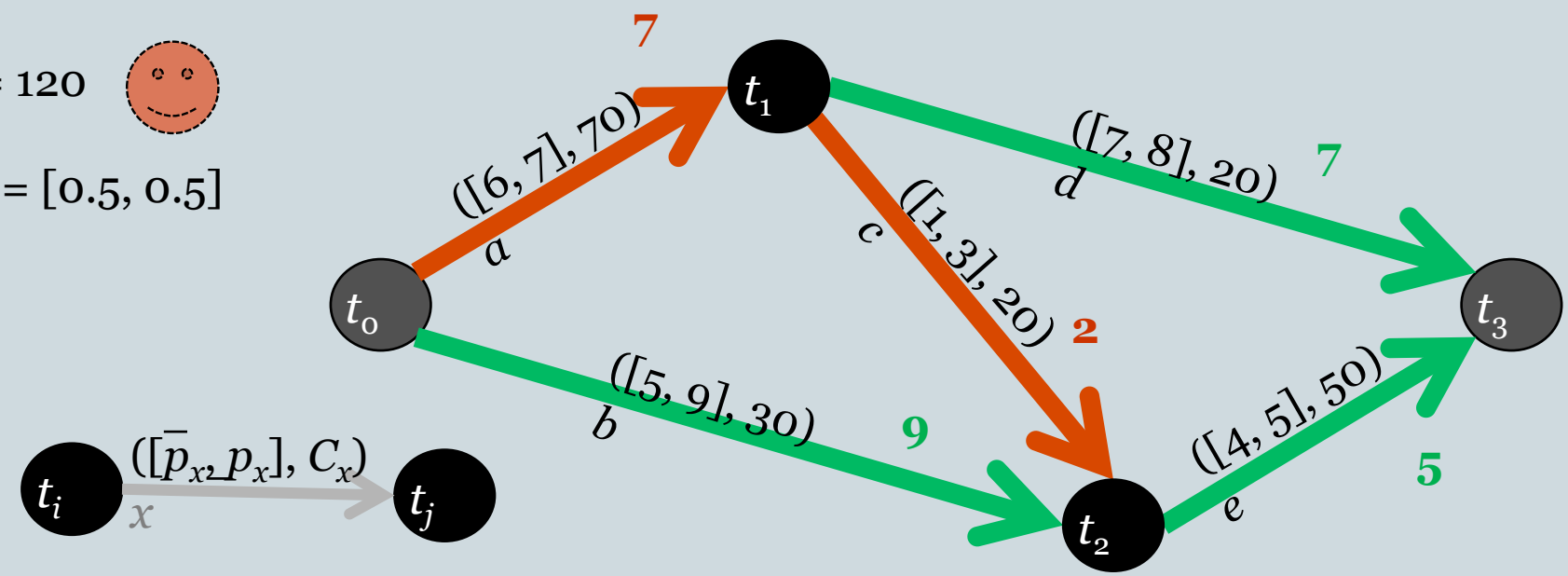


Example (inspired from Phillips and Dessouky, 1977)

t_3-t_0	Rewards	$Cost_g$	$Cost_r$	Z_g	Z_r	
15	0	0	0	0	0	S_0
14	120	20	20	40	40	S_1

$\pi = 120$

$w_u = [0.5, 0.5]$

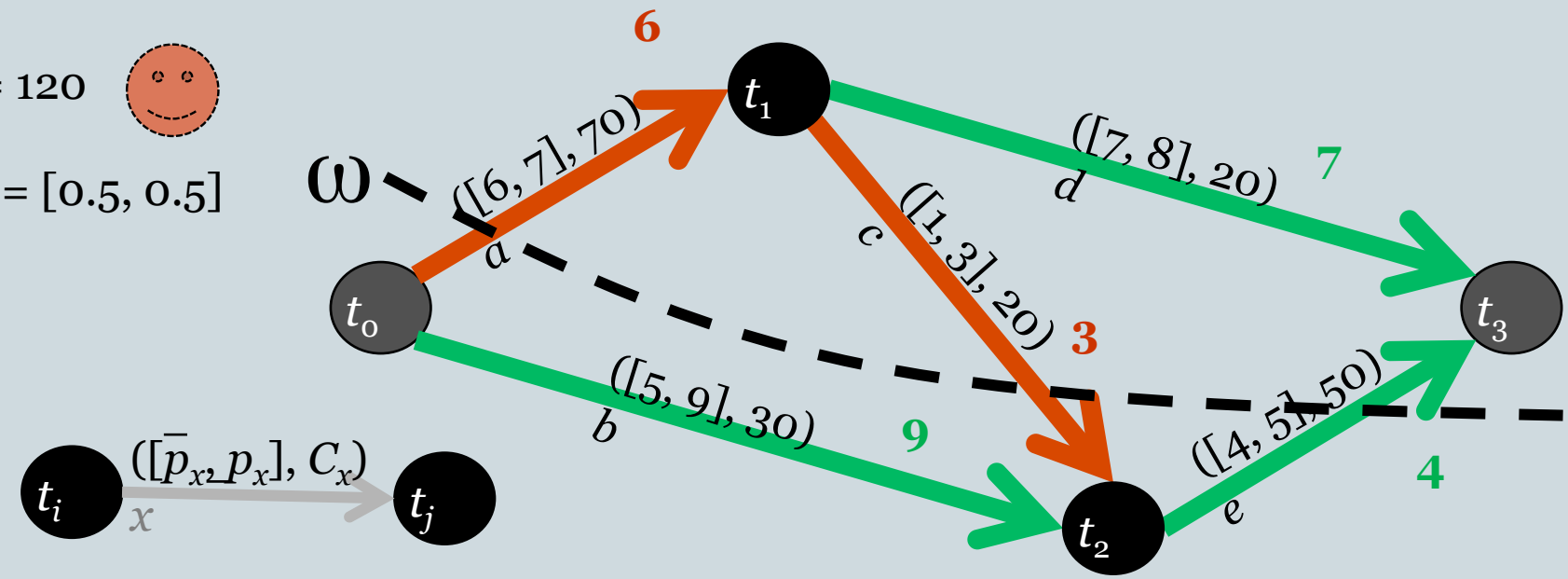


Example (inspired from Phillips and Dessouky, 1977)

t_3-t_0	Rewards	$Cost_g$	$Cost_r$	Z_g	Z_r	
15	0	0	0	0	0	S_0
14	120	20	20	40	40	S_1
13	240	70	70	50	50	S_2

$\pi = 120$

$w_u = [0.5, 0.5]$

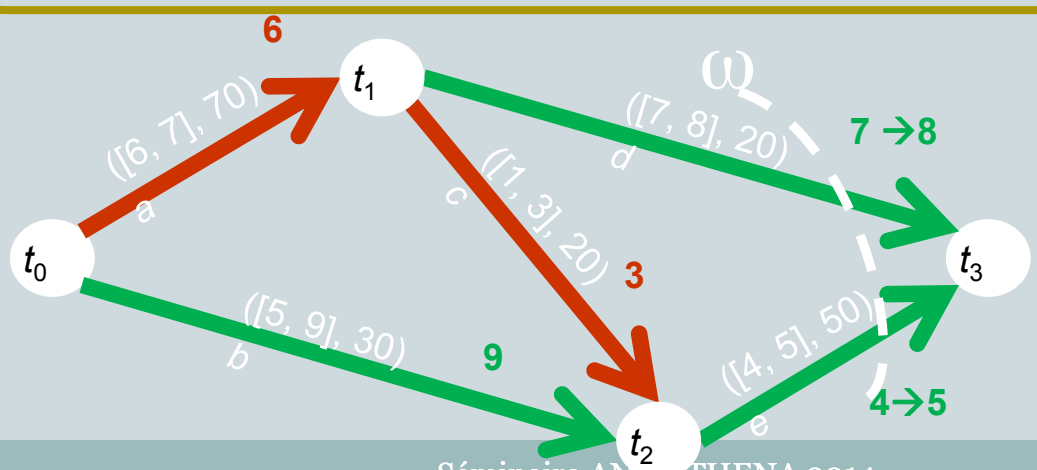


Example (inspired from Phillips and Dessouky, 1977)

t_3-t_0	Rewards	$Cost_g$	$Cost_r$	Z_g	Z_r	
15	0	0	0	0	0	S_0
14	120	20	20	40	40	S_1
13	240	70	70	50	50	S_2

The green agent can improve his profit by increasing the makespan

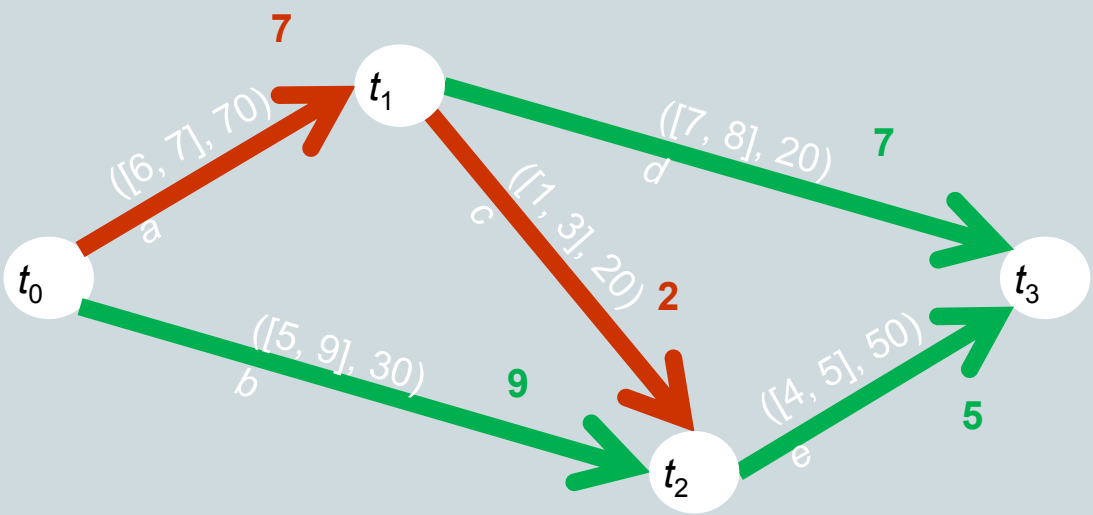
14	120	0	70	60	-10	S'_2



Example (inspired from Phillips and Dessouky, 1977)

t_3-t_0	Rewards	Cost$_g$	Cost$_r$	Z_g	Z_r	
15	0	0	0	0	0	S_0
14	120	20	20	40	40	S_1
13	240	70	50	50	70	S_2

- **S_2 is a Pareto optimum but not a Nash equilibrium**
- **S_1 is not a Pareto optimum but a Nash equilibrium**



Problem *P*: Find the best Nash equilibrium (with minimum makespan) ⇔ Price of stability

Outline

- Introduction
- Multi-Agent project scheduling (MAPS)
- <u>Finding a minimum-makespan stable strategy</u>
- Optimal reward sharing
- Conclusion

Characterization of stable strategies

(Agnetis et al., 2012)

- Non-poor strategy
  - S is poor if, leaving the makespan unchanged, an agent is able to improve its profit by itself

- A non-poor strategy *S* is a Nash equilibrium if and only if there is no residual cut ω such that
  - ω is increasing and there exists one agent such that

$$\text{profit}_u \geq w_u \pi$$

    - The agent is interested in increasing the makespan
  - ω is decreasing and there exists one agent such that

$$\text{cost}_u(\omega) < w_u \pi$$

    - Decreasing the makespan is profitable for the agent

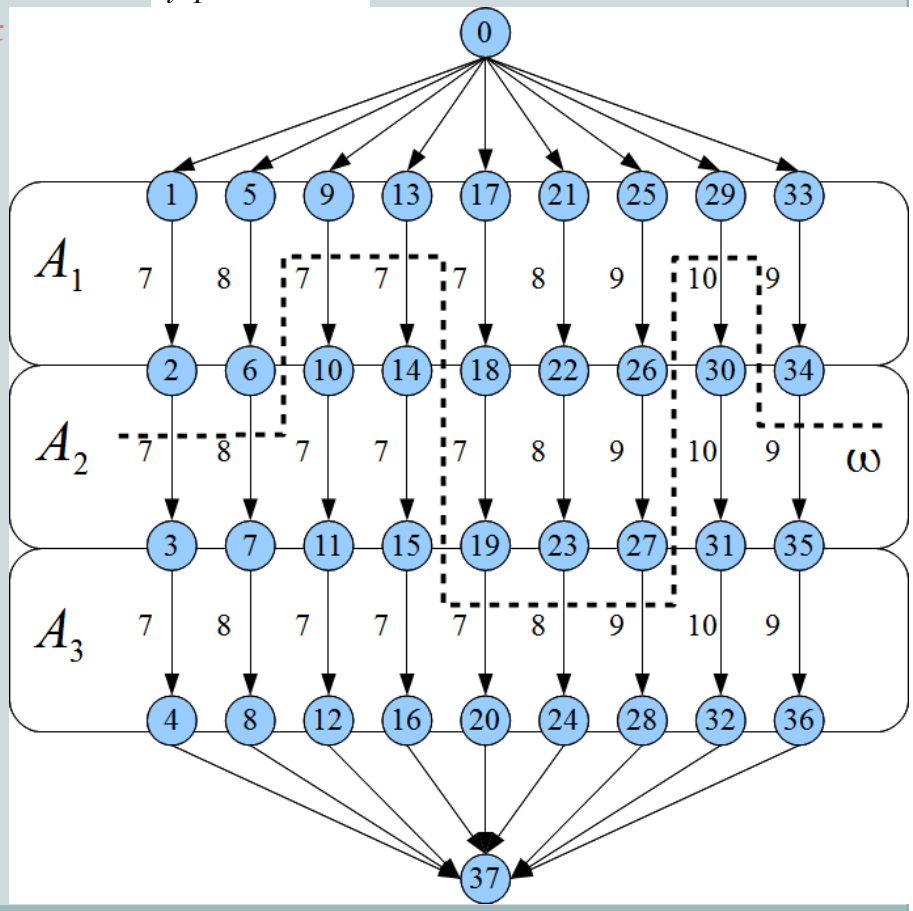
P is NP-Hard in the strong sense

(Agnetis et al., 2012)

- *3-PARTITION*
  - *A set $\{a_1, \ldots a_K\}$ of $K=3k$ positive integers such that* $\sum_{l=1}^{K} a_l = kB$ *and* $\frac{B}{4} \le a_l \le \frac{B}{2}$
  - *Q: Can it be partitioned into k subsets S_k such that* $|S_k| = 3$ *and* $\sum_{j \in S_k} a_j = B$
    - *Ex:* K=9, k=3, B=24, {7,7,7,7,8,8,9,9,10}

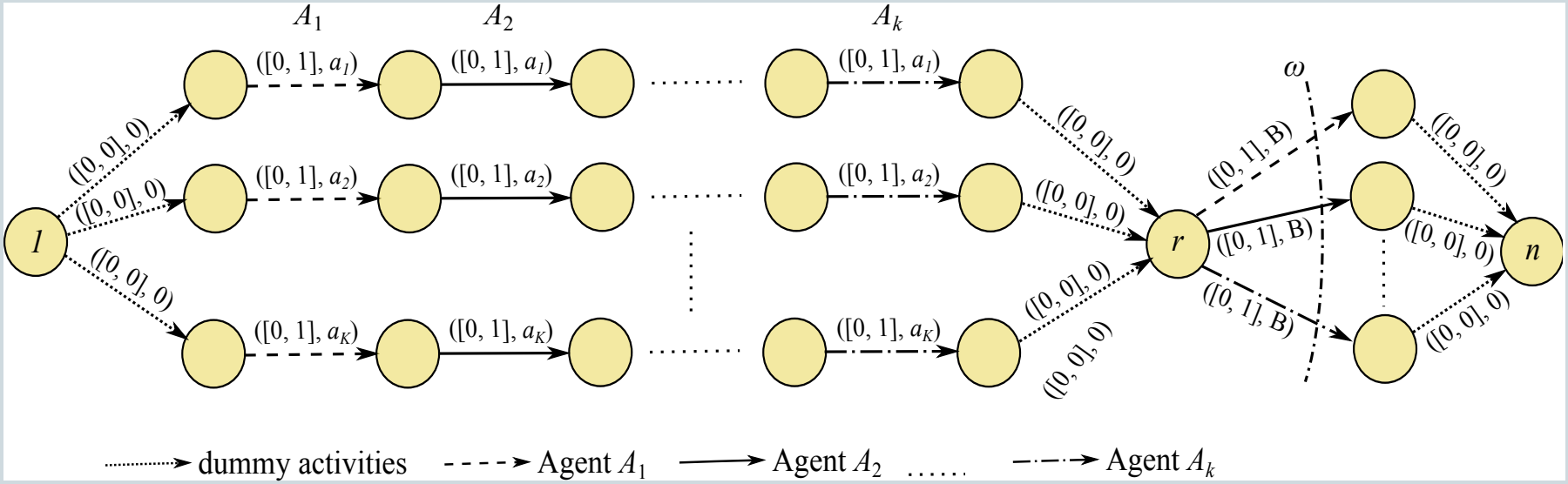
- Reduction
  - k agents in series
  - Every agent manages K activities:
    - $p_i \in [0, 1]$ and $c_i = a_i$
  - Makespan = k
  - $\pi = k(B+\varepsilon)$ and $w_u = 1/k$
  - Q: Is there a NE with makespan < k ?
    - Find a cut such that the cost for each agent does not exceed the reward $B+\varepsilon$

➔ Sol 3-PARTITION



Extension of the proof of NP-hardness to variable sharing policy (Briand et al., 2012)

- Extended Network



- ω is an optimal cut iff $w_u=1/k$
  - If there exists a strategy profile S^* and a sharing policy w_u^* such that $D(S^*)=k-1$ then the original instance of the 3-PARTITION problem is a yes instance
  - Otherwise, $D(S^*)=k$ and the original instance is a no instance

Mathematical Formulation

(Briand et al., 2012)

- What is the minimum-makespan NE ?

$$\min t_n - t_1$$

s.t.

$$t_i - t_j + p_{i,j} \leq 0 \qquad \forall (i,j) \in U$$

$$profit_u(\omega) < w_u \pi \qquad \forall A_u \in A, \forall \omega \in \Omega(S)$$

where

$$t_i \in \mathrm{R} \qquad \forall i \in X$$

$$p_{i,j} \in [\underline{p}_{i,j}, \overline{p}_{i,j}] \qquad \forall (i,j) \in U$$

project makespan

Precedence constraints

Stability constraints

How to formulate the stability constraints?

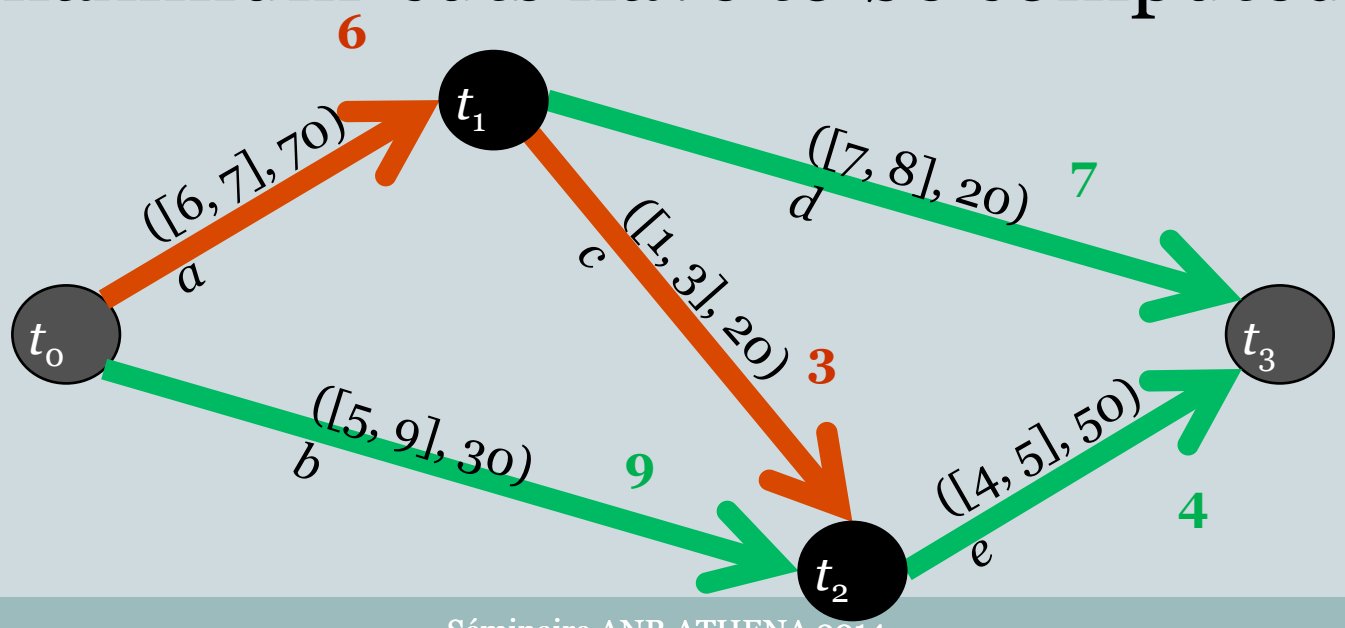
MILP Model

(Briand et al., 2012)

- Constraint reformulation is based on identification of cut $\omega \in \Omega(S)$ having maximal $\text{profit}_u(\omega)$

$$\text{profit}_u(\omega) < w_u \pi \quad \forall \omega \quad \Rightarrow \quad \max_{\forall \omega \in \Omega} \text{profit}_u(\omega) < w_u \pi$$

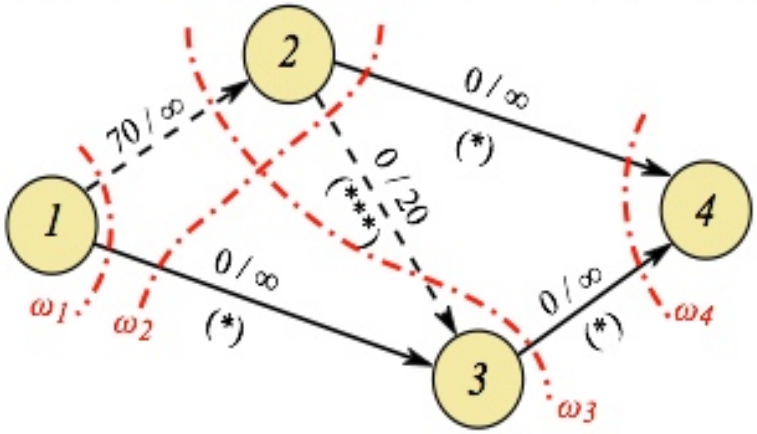
- m maximum-cuts have to be computed



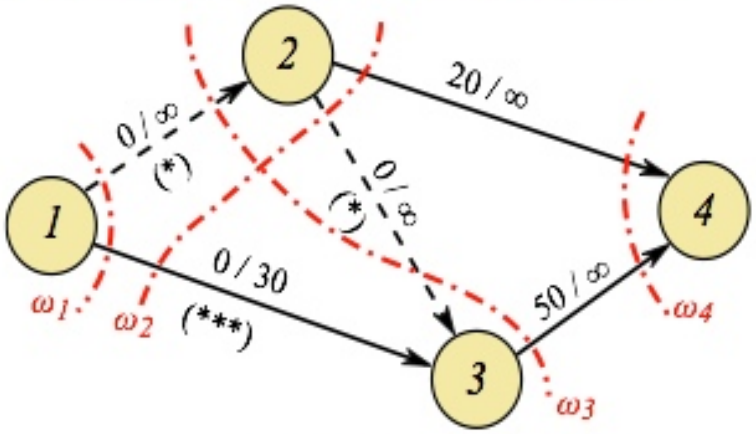
MILP Model

(Briand et al., 2012)

(*) does not belong to current agent, (**) is not critical, (***) has reached its maximal duration.



(a) $\mathcal{N}_1(S'')$ for agent A_1



(b) $\mathcal{N}_2(S'')$ for agent A_2

<u>Cost of cuts</u>:

$\text{cost}_1(\omega_1) = (70+0) = 70$

$\text{cost}_1(\omega_2) = (0+0+0) = 0$

$\text{cost}_1(\omega_3) = (70+0) - (20) = 50$

$\text{cost}_1(\omega_4) = (0+0) = 0$

Maximal cut cost $= 70 \geq \pi w_1 (= 60)$

$\Downarrow$

One profitable increasing-cut is found !

<u>Cost of cuts</u>:

$\text{cost}_2(\omega_1) = (0+0) = 0$

$\text{cost}_2(\omega_2) = (0+0+20) = 20$

$\text{cost}_2(\omega_3) = (0+50) - (\infty) = -\infty$

$\text{cost}_2(\omega_4) = (20+50) = 70$

Maximal cut cost $= 70 \geq \pi w_2 (= 60)$

$\Downarrow$

One profitable increasing-cut is found !

MILP Model

(Briand et al., 2012)

- Considering a given strategy, computing the maximum cuts is easy (slave problem)

$$
\begin{array}{ll}
\max & \sum_{(i,j)\in\mathcal{T}_u} \alpha^u_{i,j}\underline{c}^u_{i,j} - \sum_{(i,j)\in\mathcal{T}_u} \beta^u_{i,j}\overline{c}^u_{i,j} \\
\text{s.t} & \alpha^u_{i,j} - \beta^u_{i,j} - \gamma^u_i + \gamma^u_j \leq 0 \\
& \gamma^u_0 = 1, \gamma^u_{n-1} = 0 \\
& \gamma^u_i \geq 0 \\
& \alpha^u_{i,j}, \beta^u_{i,j} \geq 0
\end{array}
$$

The constraint matrix is TU

The slave problem can be embedded inside the master one using primal/dual constraints

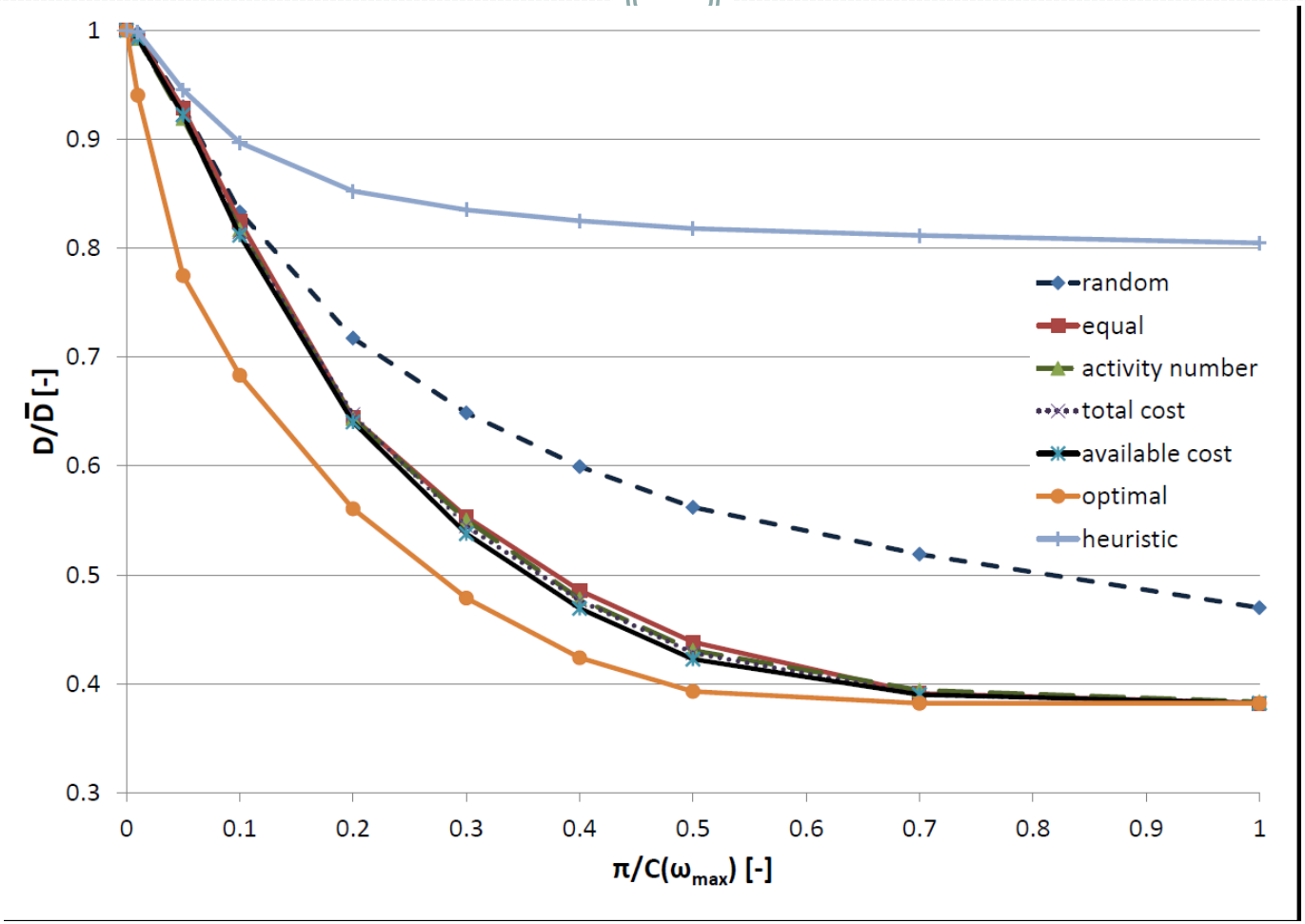
Outline

- Introduction
- Multi-Agent project scheduling (MAPS)
- Finding a minimum-makespan stable strategy
- <u>Optimal reward sharing</u>
- Conclusion

Reward sharing

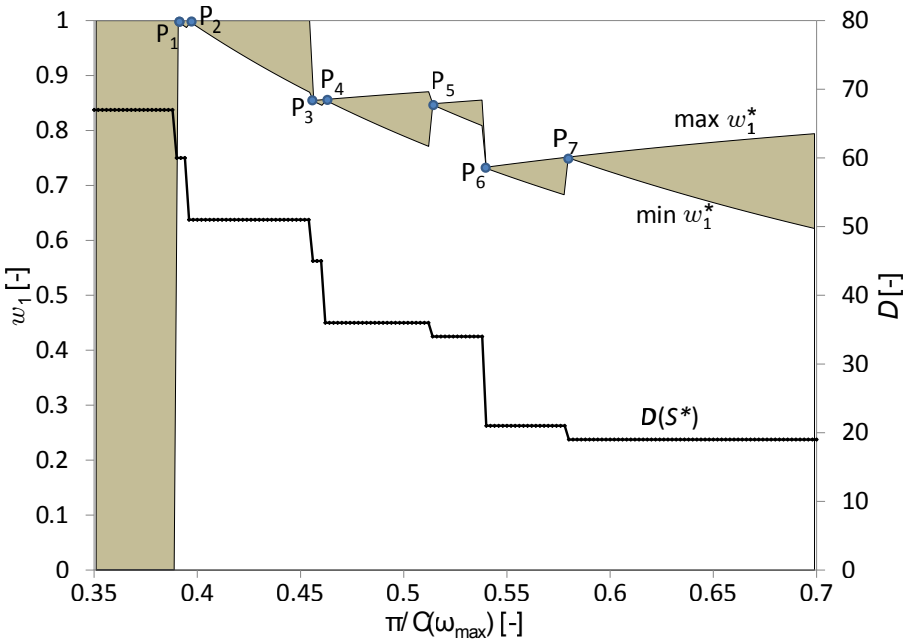
- Advantage of variable profit sharing w_u
  - From the customer viewpoint
    - What is the best reward sharing policy that offers the best (stable) makespan ?
  - From one agent's viewpoint
    - What is the best profit (stable) the agent can expect given a project makespan?
- Five fixed sharing policies have been compared with the optimal one
  - 5 rules
  - Random, Equal, Activity number, Total cost, Available cost
- The influence of π has been also tested
- Experiments have been conducted on the 60-activities instances

Sharing policies comparison

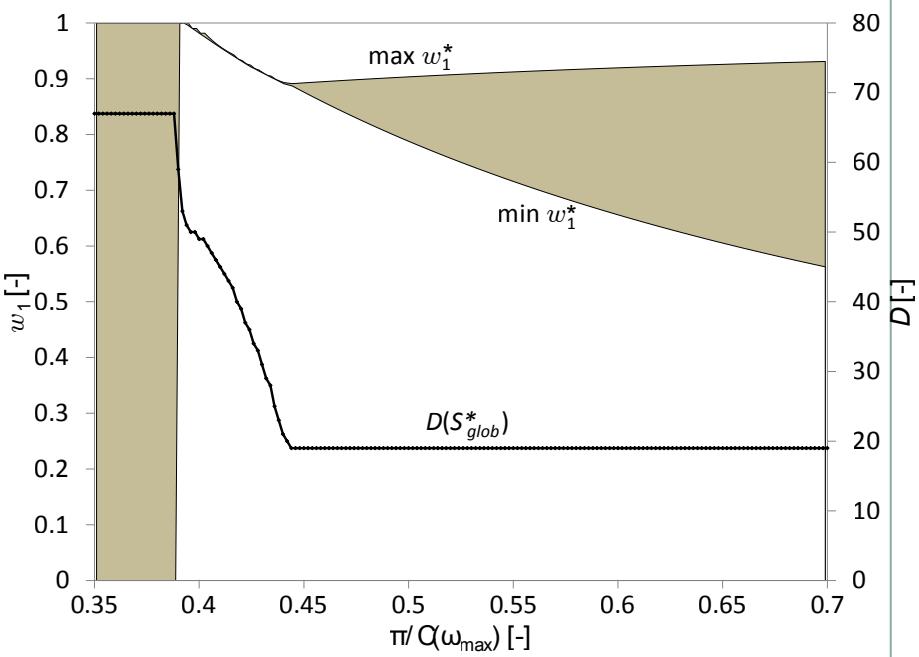


Relative makespan reduction vs the relative daily reward

Inpact of the stability constraints on the problem difficulty



(a) with the stability constraint (P)



(b) without the stability constraint

Minimum/maximum optimal profit intervals for agent A_1 vs the daily reward

Outline

- Introduction
- Multi-Agent project scheduling (MAPS)
- Finding a minimum-makespan stable strategy
- Optimal reward sharing
- <u>Conclusions</u>

Conclusions

- Multiagent project scheduling problem with controllable processing times
  - Efficiency and stability notions
  - Optimization problem = Find a Nash equilibrium and sharing policy that minimize the project makespan
    - NP-hard in the strong sense
  - MIP formulation (w_u = parameter / decision variables)
    - Comparison of sharing policies

- Stability constraint/Nash equilibrium for other combinatorial optimization problems ?

Questions?
Suggestions ?
Collaboration ?

References

- Briand, C., Ngueveu, S. U., Premysl S, 2013. *Finding an optimal Nash equilibrium to the multi-agent project scheduling problem*. (en révision). http://hal.archives-ouvertes.fr/hal-00843947

- Briand, C., Ngueveu, S. U., Premysl S, 2013. *Solving a cooperative project scheduling with controllable processing times, self-interested agents and equal profit sharing*. In Proceedings of MISTA, ISSN 2305-249X, 8 pages, Ghent, Belgium, 2013.

- Agnetis, A., Briand, C., and Billaut, J.-C, 2012. The multi-agent project scheduling problem with controllable processing times: a game theoretic approach. Rapport LAAS N°12393.