Balanced assignments of periodic tasks

Héloïse Gachet

CERMICS, École nationale des ponts et chaussées
SNCF DTIPG

December 9th, 2024

Journée GOThA

Joint work with Frédéric Meunier

Introduction

Public transports: crew scheduling and rostering with fairness between workers
$\rightarrow$ Balanced assignment of tasks to workers

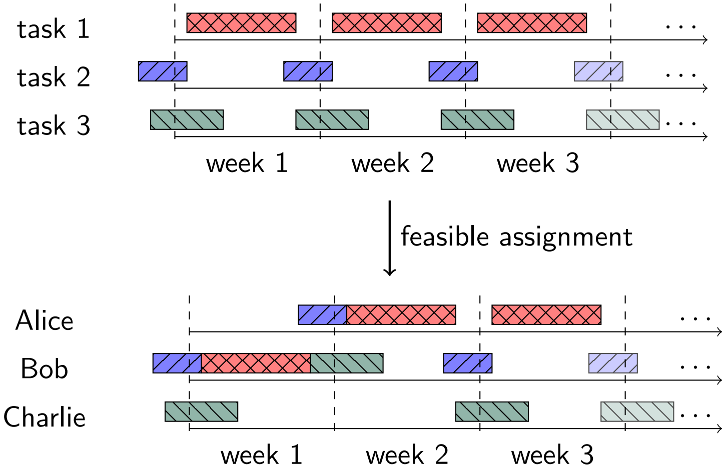
Input:

- Tasks to be performed at the same time every week
- Indistinguishable workers

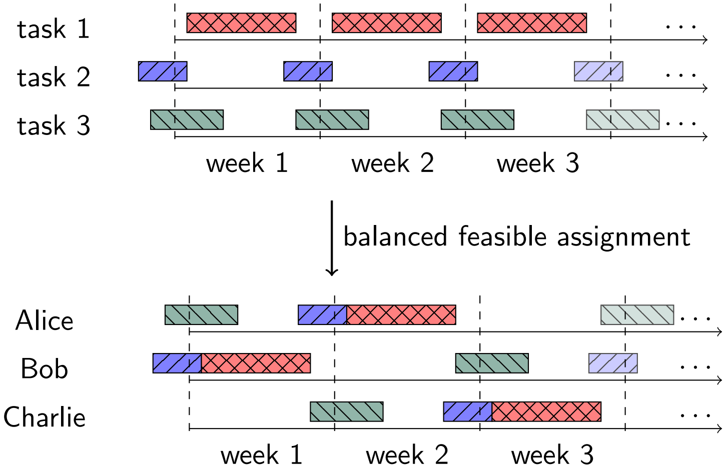
Output: Assignment of tasks to workers so that each worker performs each task with same asymptotic frequency

e.g., Monday from 9:10 to 10:30, drive train 9015 from Paris to London with Alice, Bob, or Charlie

Feasible assignment



Balanced assignment: an example



Balanced assignment: formally

Input: n tasks to be repeated every week, q workers

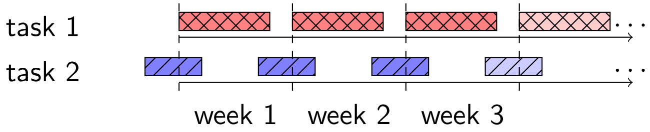
Assignment: $f\colon [n] \times \mathbb{Z}_{>0} \to [q]$, where $f(i,r) = j$ means worker j performs task i on week r

Feasible assignment: f is non overlapping

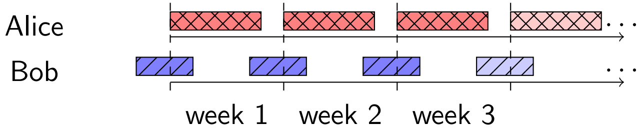
Balanced assignment:

$$\lim_{t \to +\infty} \frac{1}{t} \left|\{r \in [t] \colon f(i,r) = j\}\right| = \frac{1}{q} \qquad \forall i, j$$

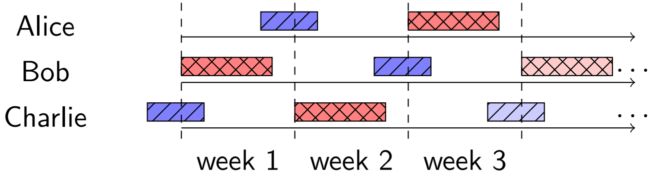
Balanced assignments: unachievable case



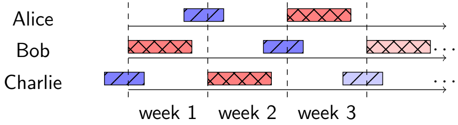
With 2 workers



With 3 workers



Main results



Theorem G.–Meunier 2024+

There exists a balanced feasible assignment if and only if there exists a feasible assignment with a worker performing each task at least once.

- Existence of a balanced feasible assignment with period q (number of workers)
- Deciding the existence of a balanced feasible assignment: polynomial problem
- Building a balanced feasible assignment: polynomial problem when q is bounded

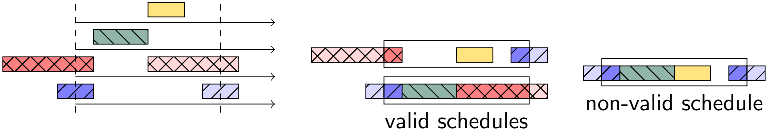
Extension

Input: n tasks to be repeated every week, q workers, $\mathcal{S}$ set of valid weekly schedules

ex. All workers perform at most M tasks per week:

$$\mathcal{S} = \left\{ S : n^{tasks}(S) \leqslant M \right\}$$

With $M = 2$



Feasible assignment: f is non overlapping AND uses only allowed weekly schedules

Extension: results

Theorem G.–Meunier 2024+

If there exists a feasible assignment with a worker performing each task at least once and all workers are busy at time zero, then there exists a balanced feasible assignment.

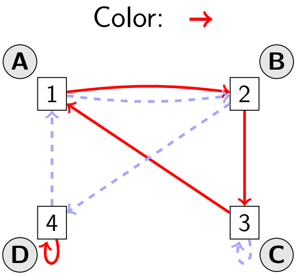
- Existence of a periodic balanced feasible assignment with large period
- Examples where the additionnal condition is needed
- Deciding the existence of a balanced feasible assignment: open

Moving pebbles

Consider:

- graph $D = (V, A)$ with colored arcs, each color forming vertex disjoint cycles covering V
- one pebble on each vertex

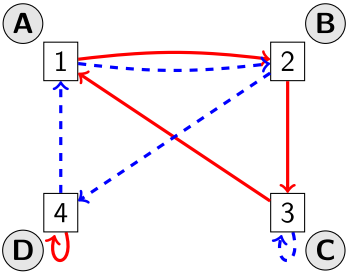
Sequence of colors defines sequence of moves of the pebbles



- Vertex = task at time zero
- Arc = valid schedule
- Pebble = worker
- Color c = partition of the tasks into schedules

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



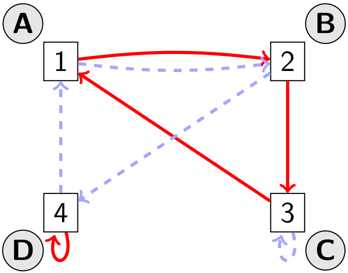
sequence of colors:

arcs visiting:

	1→2	2→3	3→1	4→4	1⇢2	2⇢4	3⇢3	4⇢1
A	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0	0
D	0	0	0	0	0	0	0	0

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



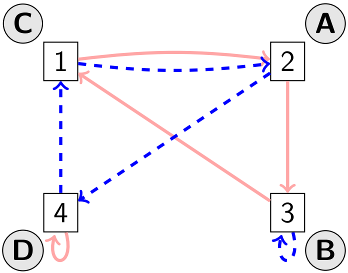
sequence of colors:

arcs visiting:

	1→2	2→3	3→1	4→4	1⇢2	2⇢4	3⇢3	4⇢1
A	**1**	0	0	0	0	0	0	0
B	0	**1**	0	0	0	0	0	0
C	0	0	**1**	0	0	0	0	0
D	0	0	0	**1**	0	0	0	0

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



sequence of colors:

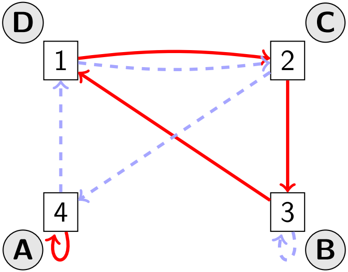
→ -→ → -→ -→ → -→ →

arcs visiting:

	1→2	2→3	3→1	4→4	1-→2	2-→4	3-→3	4-→1
A	1	0	0	0	0	**1**	0	0
B	0	1	0	0	0	0	**1**	0
C	0	0	1	0	**1**	0	0	0
D	0	0	0	1	0	0	0	**1**

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



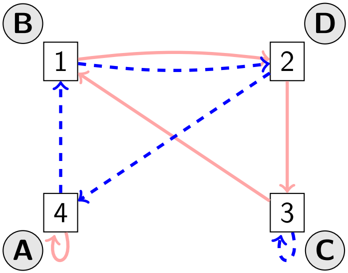
sequence of colors:

arcs visiting:

	1→2	2→3	3→1	4→4	1⇢2	2⇢4	3⇢3	4⇢1
A	1	0	0	**1**	0	1	0	0
B	0	1	**1**	0	0	0	1	0
C	0	**1**	1	0	1	0	0	0
D	**1**	0	0	1	0	0	0	1

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



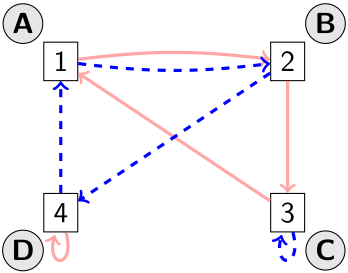
sequence of colors:

arcs visiting:

	1→2	2→3	3→1	4→4	1⇢2	2⇢4	3⇢3	4⇢1
A	1	0	0	1	0	1	0	**1**
B	0	1	1	0	**1**	0	1	0
C	0	1	1	0	1	0	**1**	0
D	1	0	0	1	0	**1**	0	1

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



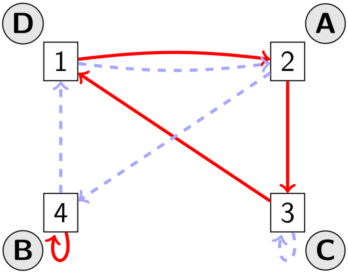
sequence of colors:

arcs visiting:

	1→2	2→3	3→1	4→4	1⇢2	2⇢4	3⇢3	4⇢1
A	1	0	0	1	**1**	1	0	1
B	0	1	1	0	1	**1**	1	0
C	0	1	1	0	1	0	**2**	0
D	1	0	0	1	0	1	0	**2**

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



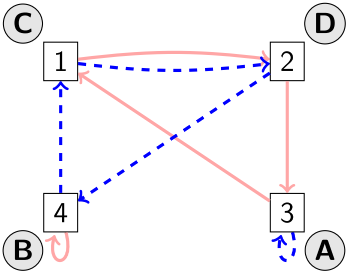
sequence of colors:

arcs visiting:

	1→2	2→3	3→1	4→4	1-→2	2-→4	3-→3	4-→1
A	1	**1**	0	1	1	1	0	1
B	0	1	1	**1**	1	1	1	0
C	0	1	**2**	0	1	0	2	0
D	**2**	0	0	1	0	1	0	2

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



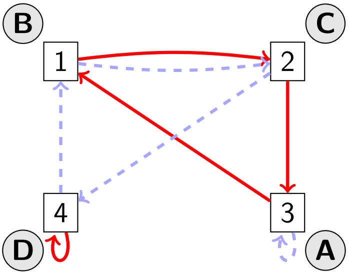
sequence of colors:

arcs visiting:

	1→2	2→3	3→1	4→4	1⇢2	2⇢4	3⇢3	4⇢1
A	1	1	0	1	1	1	**1**	1
B	0	1	1	1	1	1	1	**1**
C	0	1	2	0	**2**	0	2	0
D	2	0	0	1	0	**2**	0	2

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



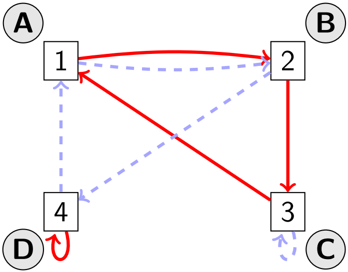
sequence of colors:

arcs visiting:

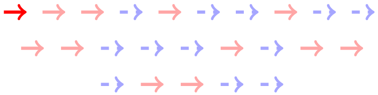
	1→2	2→3	3→1	4→4	1⇢2	2⇢4	3⇢3	4⇢1
A	1	1	**1**	1	1	1	1	1
B	**1**	1	1	1	1	1	1	1
C	0	**2**	2	0	2	0	2	0
D	2	0	0	**2**	0	2	0	2

Moving pebbles

Aim: Find a sequence of colors making each pebble visit each arc with same asymptotic frequency.



sequence of colors:



arcs visiting:

	1→2	2→3	3→1	4→4	1⇢2	2⇢4	3⇢3	4⇢1
A	**1**	0	0	0	0	0	0	0
B	0	**1**	0	0	0	0	0	0
C	0	0	**1**	0	0	0	0	0
D	0	0	0	**1**	0	0	0	0

Back to the results

Theorem G.–Meunier 2024+

If the graph is strongly connected, then there exists a sequence of colors making each pebble visit each arc with same asymptotic frequency.

2 proofs:

- Markov chains: non constructive
- Contructive proof with additionnal result:

Proposition G.–Meunier 2024+

The constructed sequence of colors is periodic with period bounded by $q^2 q!$.

q: number of workers

Thank you