

A Case Study in Workflow Scheduling Driven by Log Data

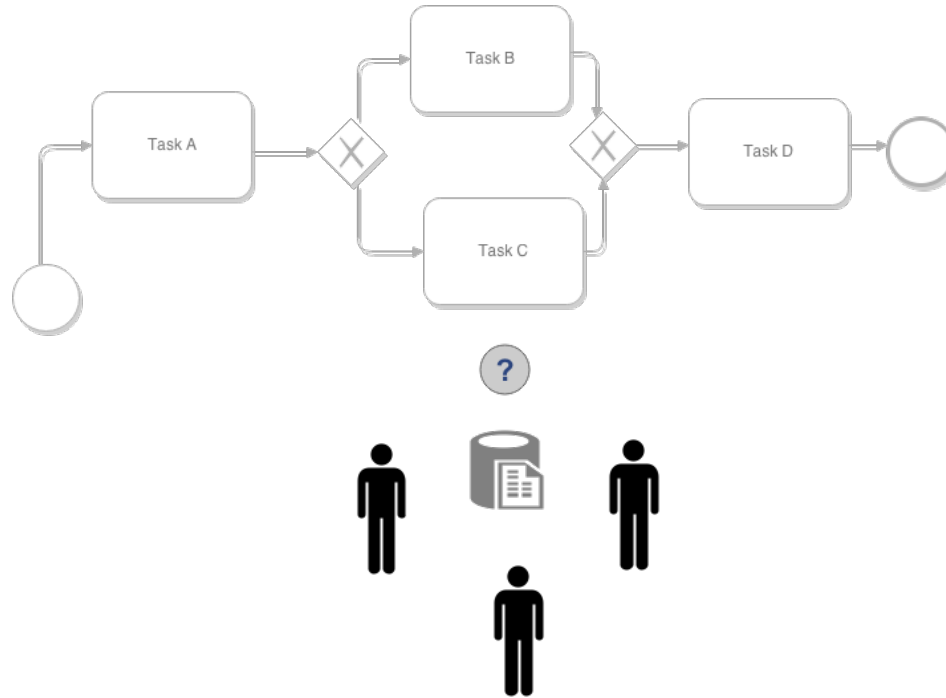
Mirela Botezatu – IBM Research – Zurich, Switzerland

Hagen Voelzer – IBM Research – Zurich, Switzerland

Remco Dijkman – Eindhoven University of Technology, The Netherlands

8th September 2014, Eindhoven, The Netherlands

What is data driven scheduling?



Outline

- Overview
 - Why data driven scheduling
 - Scheduling approaches: qualitative, quantitative
- A concrete scheduling setting
- Case study
- Experiments
- Results
- Final remarks

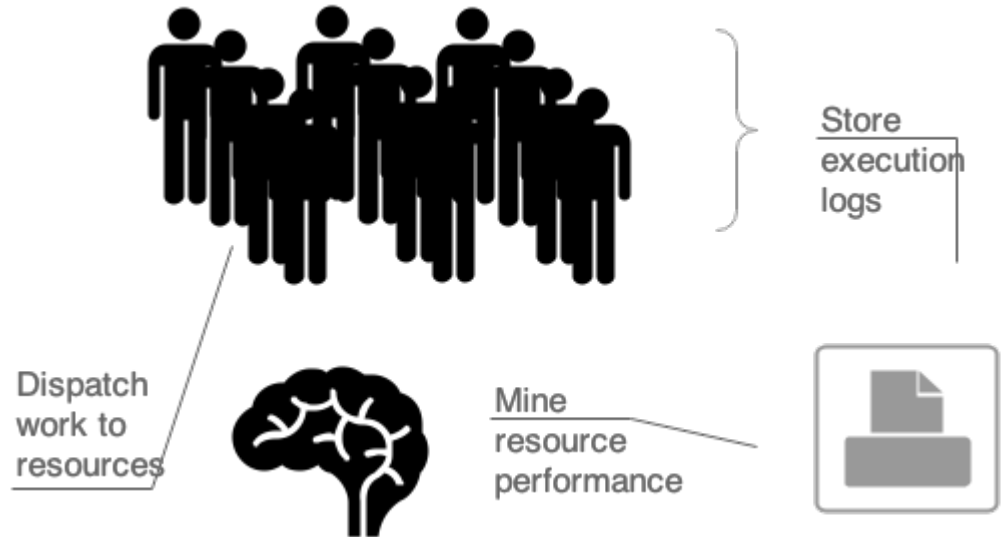
Overview

Why data driven scheduling

Keep costs down

Ensure timely completion of tasks

Keep resources load balanced



Qualitative approaches, quantitative approaches

Qualitative approaches - keywords:

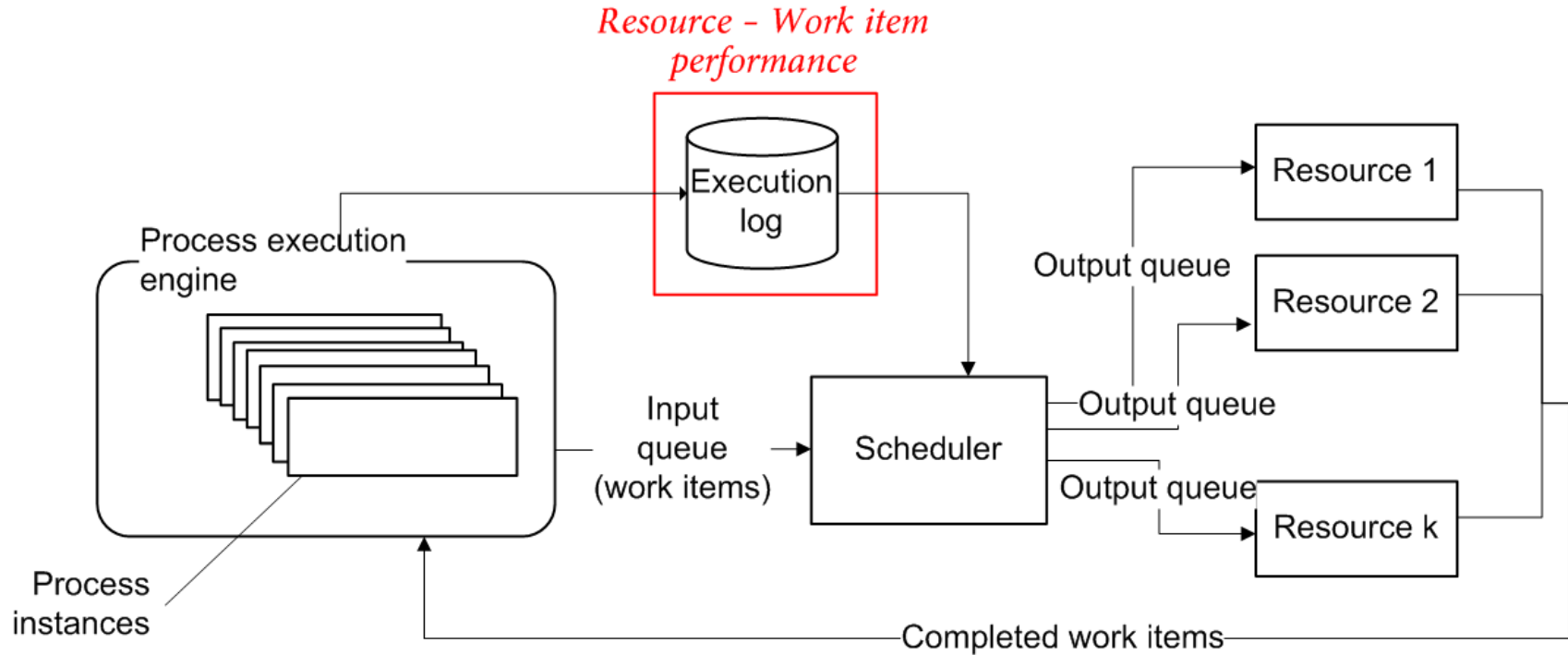
- availability
- role match
- authorization
- ...

Quantitative approaches - keywords: ← Our focus

- cost
- speed
- load
- ...

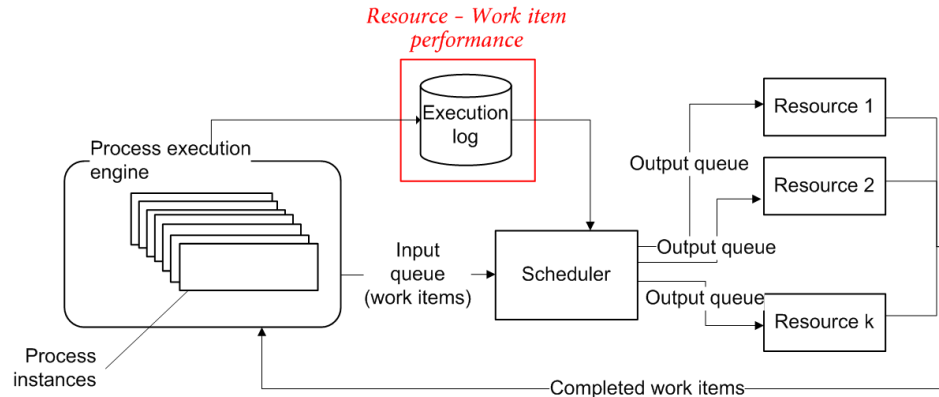
A concrete scheduling setting

Setting



Information available to the scheduler

- 1) The set of resources available on a particular day
- 2) Scheduler's input queue
- 3) State of the resource's queues
- 4) Skill/authorization information
- 5) Resource performance information



Case study

**Is quantitative scheduling useful?
Can data help in guiding the
scheduling decisions?**

Data and experimental set-up

Data:

- 262200 events in 13087 cases collected along 6 months

Assumptions:

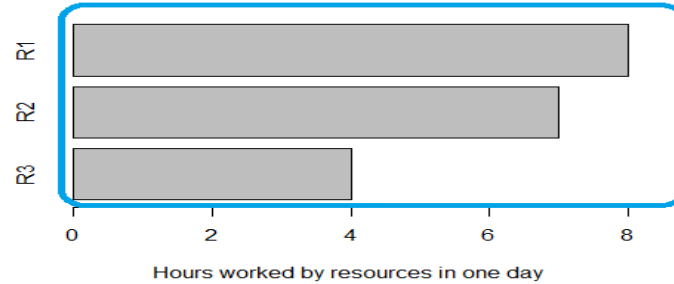
1. Maximum 8 hours of work per day for each resource
2. No work items are passed from one day to another.
3. Waiting time is not considered in any of the scheduling policies

Scheduling policies

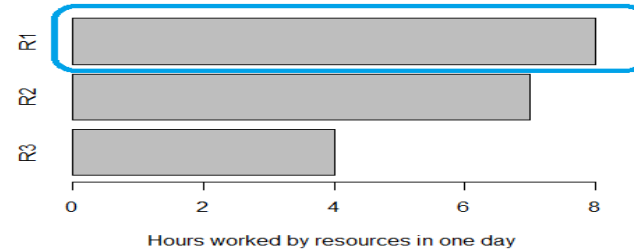
- **Greedy exploit best resource** - pick most efficient resource
- **Workload balancing** - pick least occupied resource

Cost measures

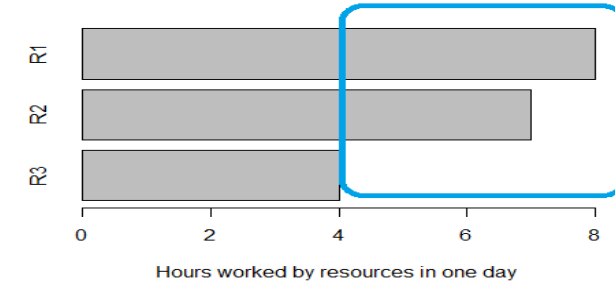
- Service time



- Makespan



- Variance in the resource load

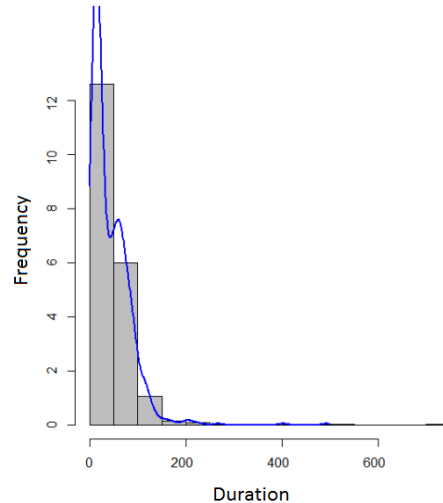


Experiments

Median duration matrix

$$\delta_{r,w} = \begin{pmatrix} \delta_{1,1} & \delta_{1,2} & \cdots & \delta_{1,k} \\ \delta_{2,1} & \delta_{2,2} & \cdots & \delta_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ \delta_{n,1} & \delta_{n,2} & \cdots & \delta_{n,k} \end{pmatrix}$$

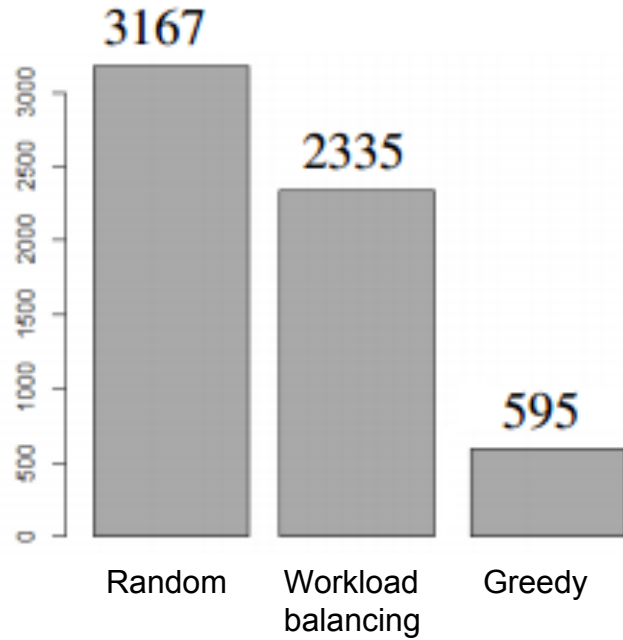
$$\delta_{r,w} = \begin{cases} x \in \mathbb{N}, & \text{resource } r \text{ is authorized to work on} \\ & \text{work item type } w \\ \text{Null,} & \text{otherwise.} \end{cases}$$



Results

Results

Service time:

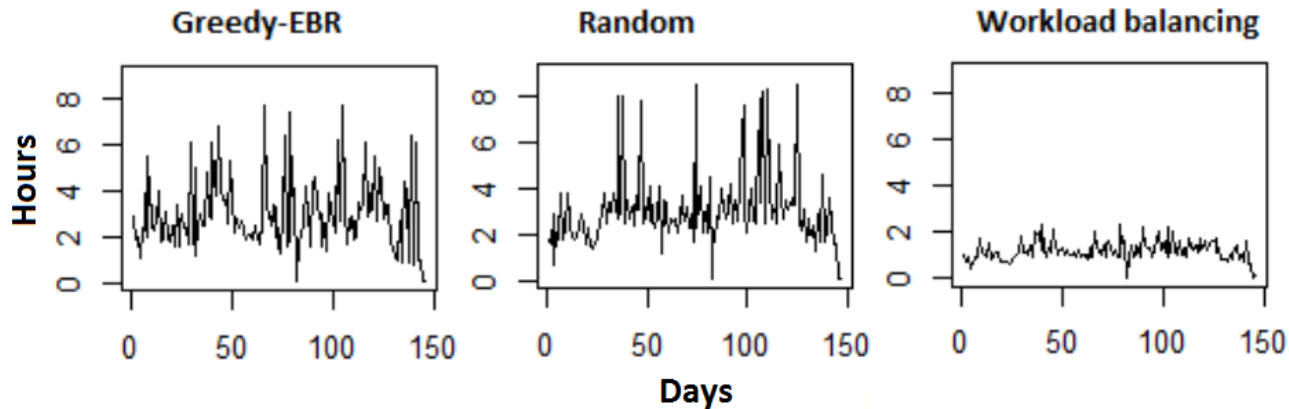


Results

Median makespan:

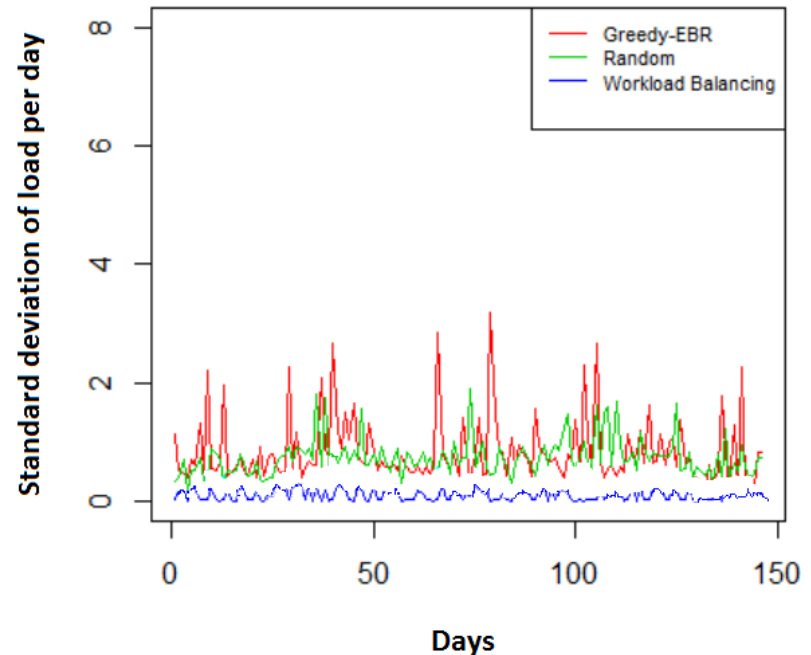
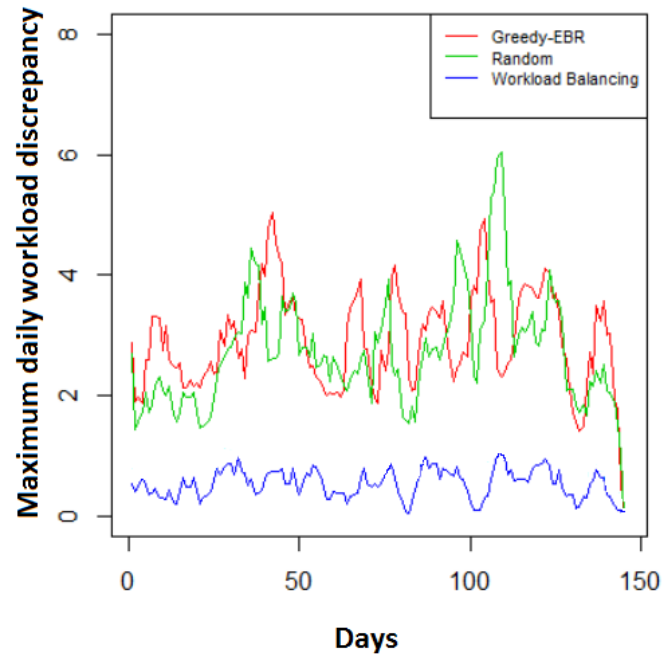
	Makespan
Greedy-EBR	2.64
WL Balancing	1.09
Random	2.8

Variance of the makespan in time



Results

Variance in the resource load:



A hybrid policy and its variations

For the hybrid policy we defined the notion of suitability:

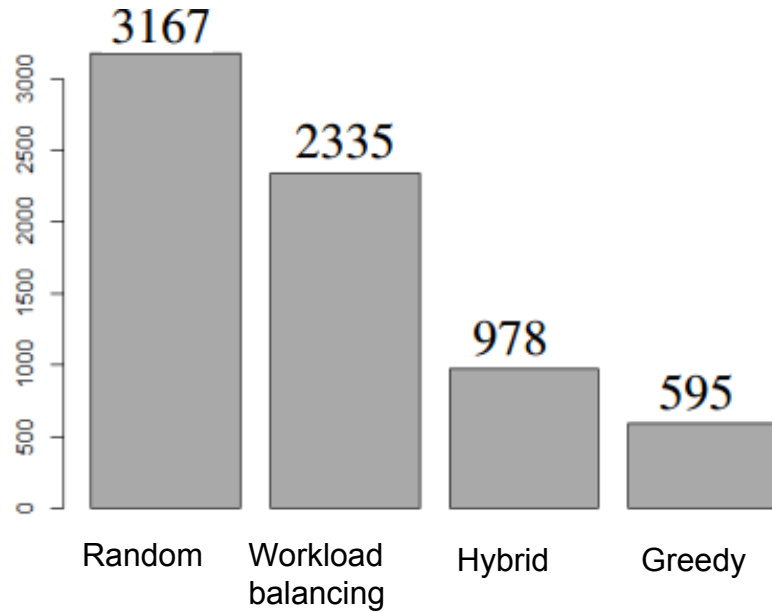
$$s_{r,i} = \delta_{r,w_i} * l_{r,i}$$

Variations of the hybrid policy:

	Load	Performance
Square root	$s_{r,i} = \delta_{r,w_i} * \sqrt{l_{r,i}}$	$s_{r,i} = \sqrt{\delta_{r,w_i}} * l_{r,i}$
Logarithm	$s_{r,i} = \delta_{r,w_i} * \log(l_{r,i})$	$s_{r,i} = \log(\delta_{r,w_i}) * l_{r,i}$

Results

Service time:

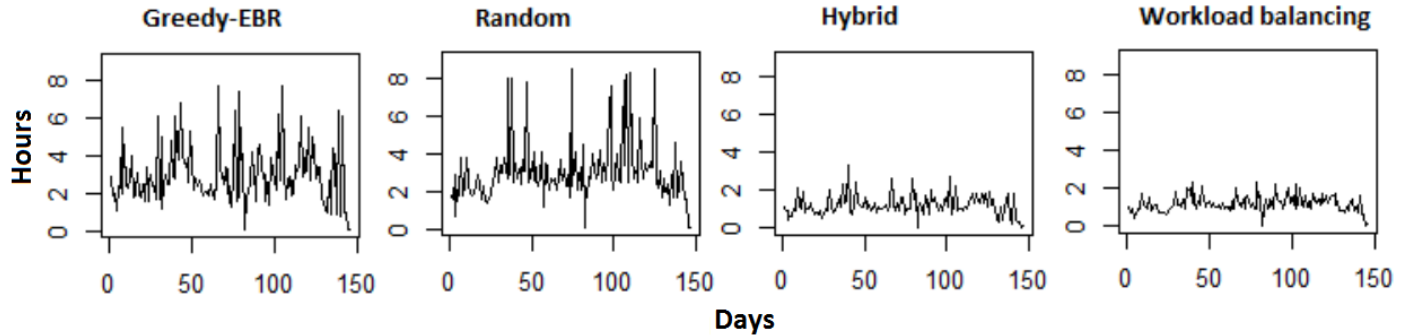


Results

Median makespan:

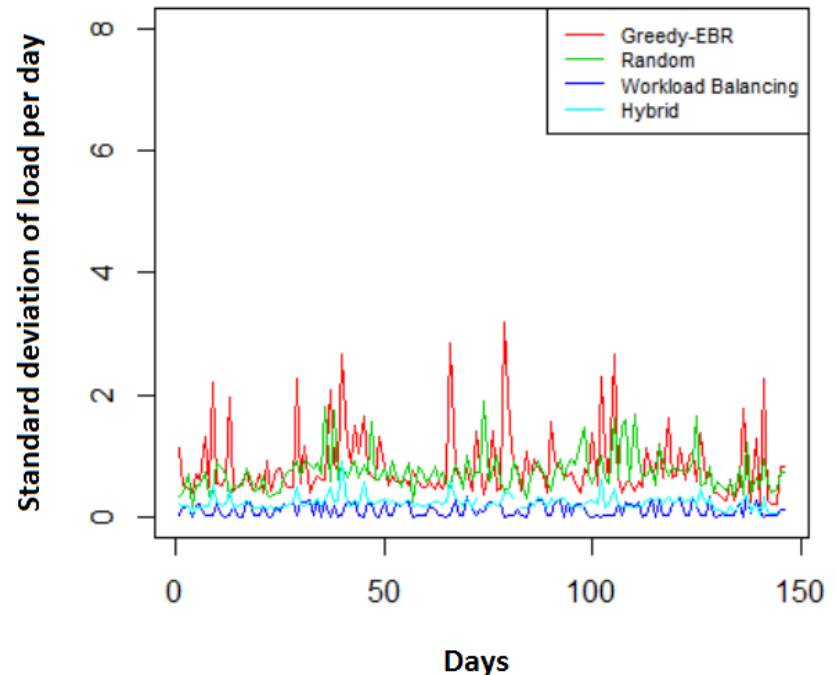
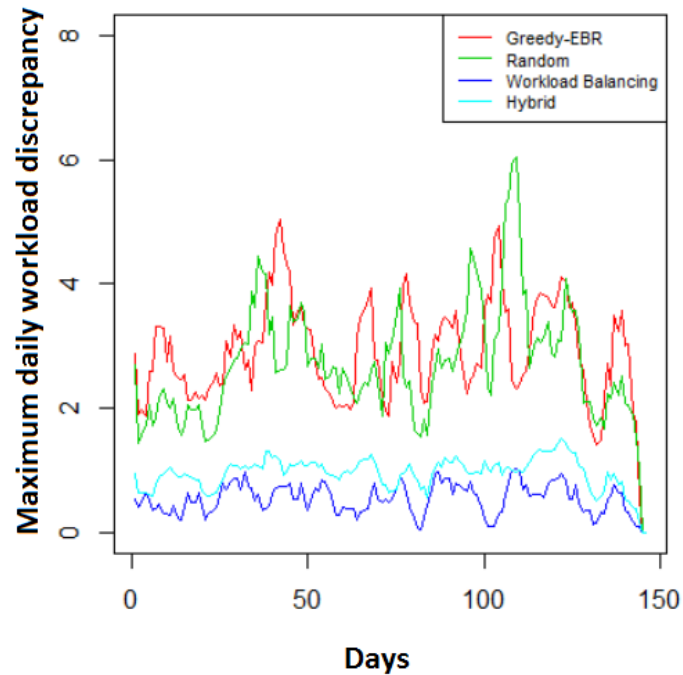
	Makespan
Greedy-EBR	2.64
WL Balancing	1.09
Hybrid	1.10
Random	2.8

Variation of the makespan in time:



Results

Variance in the resource load:



Results

Variations of the hybrid policy:

	Service time	Makespan
Hybrid (square root load)	798	1.58
Hybrid (square root performance)	1140	0.81
Hybrid (log load)	712	2.2
Hybrid (log performance)	1280	0.67
Hybrid	978	1.1

Final remarks

Conclusions

- Scheduling policies are essential for business performance
- Log data should be leveraged for accurately informing the scheduling policies
- Service time: greedy policy → 5x improvement compared to the random policy
- Makespan: workload balancing policy → 2.7x improvement compared to the random
- Tradeoff: service time - makespan /discrepancies in resource load
- Hybrid policy -> a policy that can trades-off conflicting goals

Future work

- Embed resource preference on working on a task
- More accurate estimates for resource performance based on the log data
- Trend/seasonal variation in resource performance