Scheduling unreliable jobs on parallel machines

Alessandro Agnetis, **<u>Paolo Detti</u>**, **Marco Pranzo**
Università di Siena

Manbir S. Sodhi
University of Rhode Island

Assois, May 18-21, 2008

Overview

- The Unreliable Jobs Scheduling Problem
- Related problems and applications
- Complexity analysis
- Approximation analysis
- Computational results

Stochastic activities

- There are a number of activities (projects, jobs) to execute, and there is a probability p_i to successfully carry out activity i
- Upon failure of a job, the unit (*machine*) currently in charge of its execution can't perform any other activity

The problem

- The problem is to arrange the jobs in an order that maximizes an utility index, accounting for the possibility of machine breakdowns (due to job failure)

Application context: Unsupervised systems

- In certain highly automated systems, production (at least during the night shift) is unattended
- When planning unattended activities, the chances of failures leading to machine breakdown should be taken into account
- If a machine stops, no immediate intervention is possible

The Unreliable Jobs Scheduling Problem (UJSP)

We are given:

- ***J={1,…,n}*** job set
- a single machine or a set of ***m*** identical parallel machines
- $\boldsymbol{p_i}$ success probability of job ***i***
- $\boldsymbol{r_i}$ reward for job *i*, if completed

UJSP

- **Single machine case**

Let σ be a schedule of the n jobs on the machine, and $\sigma(j)$ be the job in j-th position

the total expected reward is

$$ER(\sigma) = p_{\sigma(1)} r_{\sigma(1)} + p_{\sigma(1)} p_{\sigma(2)} r_{\sigma(2)} + \ldots +$$
$$p_{\sigma(1)} p_{\sigma(2)} \ldots p_{\sigma(n-1)} p_{\sigma(n)} r_{\sigma(n)}$$

UJSP

- **Parallel machine case**

Let $\phi=\{\sigma_1,\ldots,\sigma_m\}$ be a schedule of the ***n*** jobs on a the ***m*** machines, where σ_k is the schedule of the jobs assigned to the ***k***-th machine

the *total expected reward* is

$$ER(\phi) = ER(\sigma_1)+ER(\sigma_2)+\ldots+ER(\sigma_m)$$

A related problem

Total Weighted Discounted Completion Time (TWDCT)

$$P/\ /\sum_i w_i e^{-rC_i}$$

n jobs must be scheduled on ***m*** parallel machines

$\boldsymbol{t_i}$ processing time of job ***i***

$\boldsymbol{w_i}$ reward of job ***i***

r>0 a fixed discount rate

$\boldsymbol{w_i\, e^{-rt}}$ reward of job ***i*** if completed at time ***t***

$\boldsymbol{C_i}$ completion time of job ***i*** in some schedule ϕ

The *present value of the total reward* (to maximize) is

$$PV(\phi) = \sum_i w_i e^{-rC_i}$$

Equivalence between TWDCT and UJSP

Given an instance of TWDCT with ***n*** jobs and ***m*** parallel machines

we build an instance of UJSP with jobs ***n*** and ***m*** machines setting:

$$\mathrm{p_i} = e^{-rt_i}$$

$$\mathrm{r_i} = \mathrm{w_i}$$

The expected reward to maximize is

$$\mathrm{ER}(\phi) = \sum_{k=1}^{m} ER(\sigma_k) = PV(\phi)$$

The Weighted Sum Completion Time (WSCT)

$$P \mid \mid \sum_i w_i C_i$$

n jobs must be scheduled on ***m*** parallel machines

t_i processing time of job ***i***

w_i weight of job ***i***

C_i completion time of job ***i*** in some schedule ϕ

The function to minimize is

$$WC(\phi) = \sum_i w_i C_i$$

WSCT is a special case of TWDCT

$$\left(P\,|\,|\sum_i w_i C_i\right) \qquad\qquad \left(P\,|\,|\sum_i w_i e^{-rC_i}\right)$$

When r<<1 we have

$$e^{-rC_i} = 1 - rC_i + \frac{(rC_i)^2}{2!} - \frac{(rC_i)^3}{3!} + \frac{(rC_i)^4}{4!} + \ldots \approx 1 - rC_i$$

Hence

$$\max PV(\phi) \approx \max \sum_i w_i (1 - C_i) \equiv \sum_i w_i - \min \sum_i w_i C_i \equiv$$

$$\equiv \min WC(\phi)$$

Single machine case

UJSP, **TWDCT** and **WSCT** problems can be optimally solved by the following ordering rules

	Index	Ordering	Ref.
UJSP	$Z_i = \frac{p_i r_i}{1 - p_i}$	nonincreasing	Mitten 1960
TWDCT	$\frac{w_i}{1 - e^{-rt_i}}$	nondecreasing	Rothkopf 1966
WSCT	$\frac{w_i}{t_i}$	nonincreasing	Smith 1956

Related problems and application contexts (single machine case)

- Component testing (Monma and Sidney, 1979)
- Data acquisition and processing problems in sensor networks (Srivastava et al., 2005).
- Management of queries in databases (Hellerstein and Stonebraker, 1993)

UJSP with identical parallel machines

UJSP with **2** parallel machine is strongly NP-hard

Approximation results for two simple heuristics when ***m***=2:

- Round Robin heuristic (**RR**)
- “Highest probability” heuristic (**HP**)

Round robin heuristic (RR)

- Order the jobs according to the ratio $Z_i = \frac{p_i r_i}{1 - p_i}$

- Assign jobs to the machines in a round robin way: To machine ***h*** are assigned jobs i<n

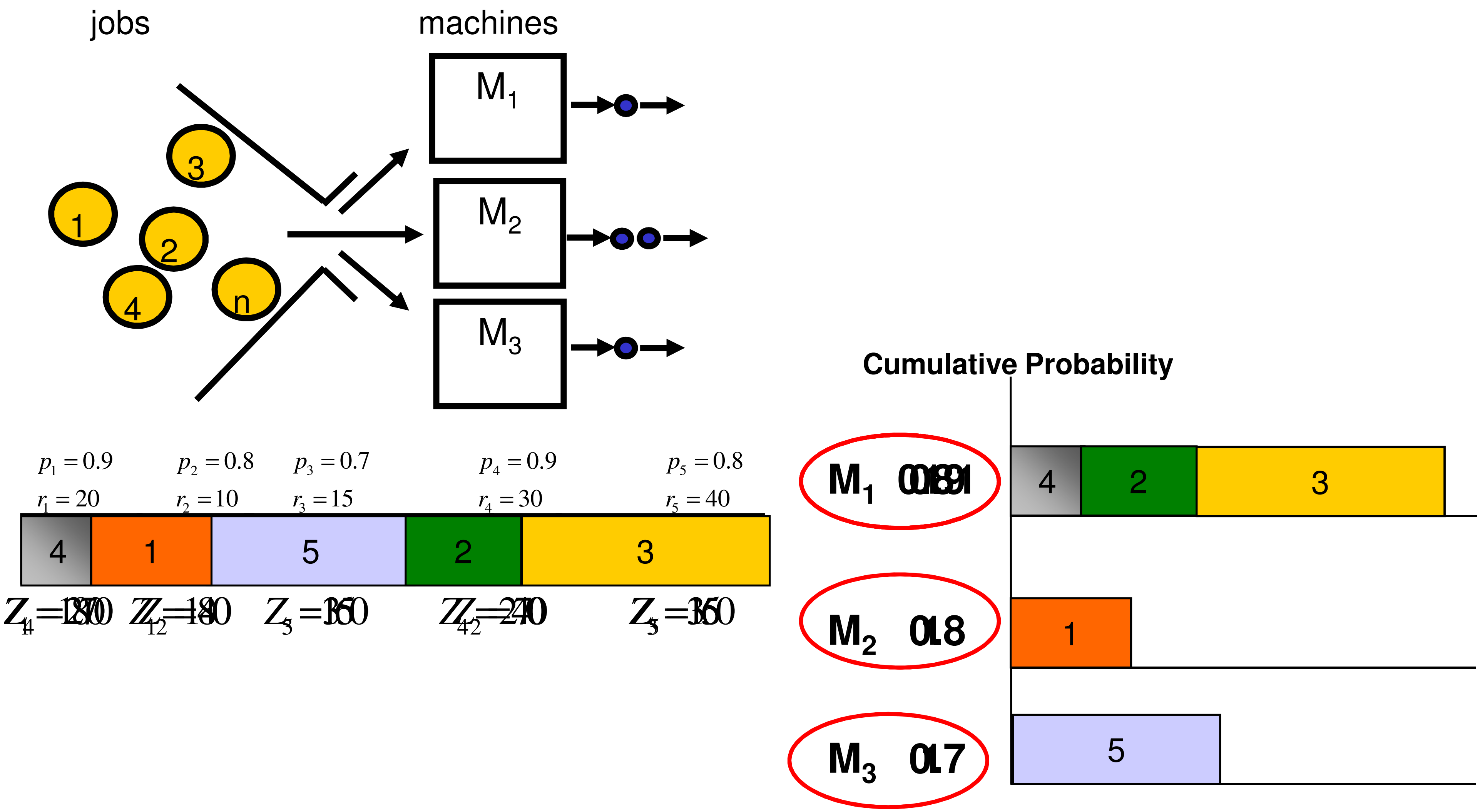
$$i = m \times k + h \ \text{ for } k = 0, 1, \ldots, \left\lfloor \frac{n}{m} \right\rfloor$$

Highest probability heuristic (HP)

- Order the jobs according to the ratio $Z_i = \dfrac{p_i r_i}{1 - p_i}$ if jobs have the same Z-ratio sequence first the job with the smallest success probability

- Assign the next job to the machine having the highest *cumulative probability*, i.e., the highest product of the probabilities of the jobs already assigned to it.

HP heuristic



An approximation result

In problem ***UJSP*** with ***m*** parallel machines, any schedule ϕ in which the jobs are sequenced according to the ratios $Z_i = \frac{p_i r_i}{1 - p_i}$ on each machine is at least

1/m-approximate.

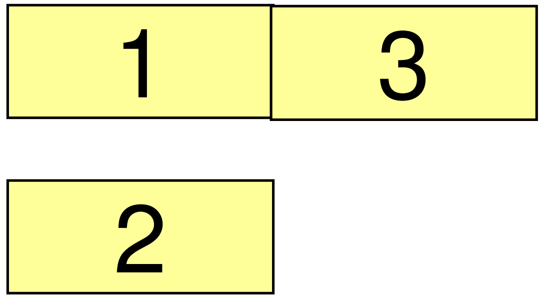
$$\frac{ER(\phi)}{ER^*} \geq \frac{1}{m}$$

How bad is RR heuristic?

A 3-job 2-machine instance

	p_i	r_i	Z_i
1	ε	$1/\varepsilon$	$1/(1-\varepsilon)$
2	$1-\varepsilon$	$\varepsilon/(1-\varepsilon)$	1
3	ε	$(1-\varepsilon)/\varepsilon$	1

Round robin solution

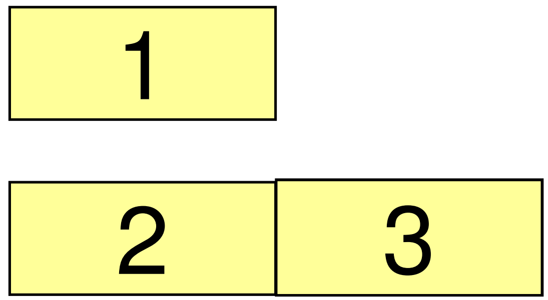


$$ER(\sigma_1{}^{RR}) = \square\, p_1 r_1 + p_1 p_3 r_3 = 1 + \varepsilon\,(1 - \varepsilon) = 1 + \varepsilon - \varepsilon$$

$$ER(\sigma_2{}^{RR}) = p_2 r_2 = \varepsilon$$

$$ER^{RR} = ER(\sigma_1{}^{RR}) + ER(\sigma_2{}^{RR}) = 1 + 2\varepsilon - \varepsilon^2$$

Optimal solution



$$ER(\sigma_1^*) = p_1 r_1 = 1$$

$$ER(\sigma_2^*) = p_2 r_2 + p_2 p_3 r_3 = \varepsilon + (1 - \varepsilon)^2 =$$

$$= \varepsilon + 1 + \varepsilon^2 - 2\varepsilon$$

$$ER^* = ER(\sigma_1^*) + ER(\sigma_2^*) = 2 - \varepsilon + \varepsilon^2$$

Approximation ratio

$$\frac{ER^{RR}}{ER^{*}} = \frac{1+2\varepsilon-\varepsilon^{2}}{2-\varepsilon+\varepsilon^{2}} \longleftrightarrow \quad 0.5$$

RR is *1/2-approximate*.

How bad is HP heuristic?

When *m*=2 HP is 0.81-approximate $\frac{ER_{HP}}{ER^*} \geq 0.81$

A first (high multiplicity) upper bound for UJSP

- A special case is when there exist several identical jobs of few different types
- If exactly ***m*** copies exist of each job, then the optimal solution consists in assigning one job of each type on each of the ***m*** machines, and sequencing according to the ***Z***-ratio

A first (high multiplicity) upper bound for UJSP

- The HM case can be exploited to devise an **upper bound** for the general case
- Given an instance of **UJSP**, replace each job ***i*** with ***m*** identical jobs ***k***:

$$p_k = \sqrt[m]{p_i} \qquad\qquad r_k = \frac{(1-\sqrt[m]{p_i})\,p_i}{(1-p_i)\sqrt[m]{p_i}}\, r_i$$

$$UB_{HM} = m\sum_{i=1}^{n} \frac{1-\sqrt[m]{p_i}}{1-p_i}\, p_i r_i \sqrt[m]{\prod_{k=1}^{i-1} p_k}$$

A first lower bound for UJSP

- The **HP** heuristic schedules at each step a job on the machine having the maximum *cumulative* probability. It can be proved that in **HP** the contribution of the ***i***-th job is *at least*

$$p_i r_i \sqrt[m]{\prod_{k=1}^{i-1} p_k}$$

- A lower bound for **HP** solution is then

$$\boxed{LB_1 = \sum_{i=1}^{n} p_i r_i \sqrt[m]{\prod_{k=1}^{i-1} p_k}}$$

A first ratio

- W.l.o.g. let ***1*** be the job with the smallest probability $\boldsymbol{p_1}$, we have

$$\frac{ER_{HP}}{ER^*} \geq \frac{LB_1}{UB_{HM}} = \frac{\sum_{i=1}^{n} p_i r_i \sqrt[m]{\prod_{k=1}^{i-1} p_k}}{m \sum_{i=1}^{n} \frac{1-\sqrt[m]{p_i}}{1-p_i} p_i r_i \sqrt[m]{\prod_{k=1}^{i-1} p_k}} \geq \frac{1}{\frac{m\left(1-\sqrt[m]{p_i}\right)}{1-p_1}}$$

New upper and lower bounds when all jobs have the same *Z*-ratio and *m*=2

- Consider a UJSP instance in which all jobs have the same ratio ***Z***
- Let $\boldsymbol{S_k}$ be the set of jobs scheduled on machine ***k***
- The expected reward on machine ***k*** is then

$$ER_k = Z\left(1 - \prod_{i \in S_k} p_i\right)$$

Expected reward of a schedule (m=2) (all jobs have the same ratio Z)

- Given a schedule $\phi=\{\sigma_1,\sigma_2\}$ for m=2, w.l.o.g. suppose that job ***1*** (with the smallest probability) is assigned to machine ***1***.
- let p_1P_A and P_B be the cumulative probabilities of jobs assigned to machine ***1*** and ***2***, respectively.
- If all jobs have the same ratio Z, the total expected reward of ϕ is

$$ER(\phi) = Z(1 - p_1 P_A) + Z(1 - P_B)$$

A second lower bound (m=2) (all jobs have the same ratio Z)

- Consider the schedule $\phi'=\{\sigma_1,\sigma_2\}$ in which job ***1*** is assigned to machine ***1*** and all other jobs are assigned to machine ***2***.
- Let P'_B the product of the probabilities of all other jobs (assigned to machine ***2***). Hence, $P'_B=P_A P_B$
- Note that the **HP** heuristic produces a schedule not worse than ϕ' (ϕ' provides a lower bound to **HP** sol.)
- If all jobs have the same ratio ***Z***, a lower bound to the solution provided by **HP** is

$$\boxed{LB_2 = Z(1-p_1) + Z(1-P'_B)}$$

A second ratio (m=2)

- By definition $P'_B = P_A P_B$, hence we have:

$$\frac{ER_{HP}}{ER^*} \geq \frac{LB_2}{ER(\phi)} \geq \frac{Z(1-p_1)+Z(1-P_A P_B)}{Z(1-p_1 P_A)+Z(1-P_B)}$$

- Which is minimized when $P_A = P_B = 0$
- Hence

$$\frac{ER_{HP}}{ER^*} \geq \frac{LB2}{ER(\phi)} \geq 1-p_1/2$$

This result holds even when jobs have different Z-values

An approximation result for HP when m=2

- The minimum of maximum between

$$\frac{ER_{HP}}{ER^{*}} \geq \frac{LB1}{UB_{HM}} \geq \frac{1-p_1}{2\left(1-\sqrt{p_1}\right)}$$

and

$$\frac{ER_{HP}}{ER^{*}} \geq \frac{LB2}{ER^{*}} \geq 1-p_1/2$$

is about 0.81 (for $\boldsymbol{p_1}$=0.37)

HP is 0.81-approximate

Experimental results

- $n = 50, 100, 500$
- $m = 2, 5, 10, 20$
- $p_i \sim U[0.7, 0.99]$
- $p_i \sim U[0.3, 0.99]$
- $p_i \sim U[0.3, 0.7]$
- $r_i \sim U[10, 40]$
- 100 randomly generated instances for each setting

Average gap of RR

$$100 * \frac{UB_{HM} - ER_{RR}}{UB_{HM}}$$

jobs / Machines	$U[0.7, 0.99]$			$U[0.3, 0.99]$			$U[0.3, 0.7]$		
	n=50	**n=100**	**n=500**	**n=50**	**n=100**	**n=500**	**n=50**	**n=100**	**n=500**
2	0.126	0.072	0.027	0.258	0.147	0.046	0.312	0.210	0.101
5	0.428	0.235	0.074	0.723	0.406	0.124	1.055	0.617	0.228
10	0.760	0.457	0.123	1.541	0.799	0.219	2.053	1.173	0.382
20	1.362	0.801	0.215	3.184	1.606	0.391	$\boxed{4.177}$	2.141	0.615

Average gap of HP

$$100 * \frac{UB_{HM} - ER_{HP}}{UB_{HM}}$$

jobs / Machines	$U[0.7, 0.99]$			$U[0.3, 0.99]$			$U[0.3, 0.7]$		
	n=50	**n=100**	**n=500**	**n=50**	**n=100**	**n=500**	**n=50**	**n=100**	**n=500**
2	0.069	0.034	0.007	0.162	0.082	0.017	0.285	0.186	0.089
5	0.251	0.114	0.022	0.557	0.266	0.052	0.936	0.569	0.203
10	0.547	0.256	0.045	1.272	0.577	0.107	1.809	1.023	0.348
20	1.133	0.552	0.091	2.904	1.312	0.222	$\boxed{3.693}$	1.880	0.553