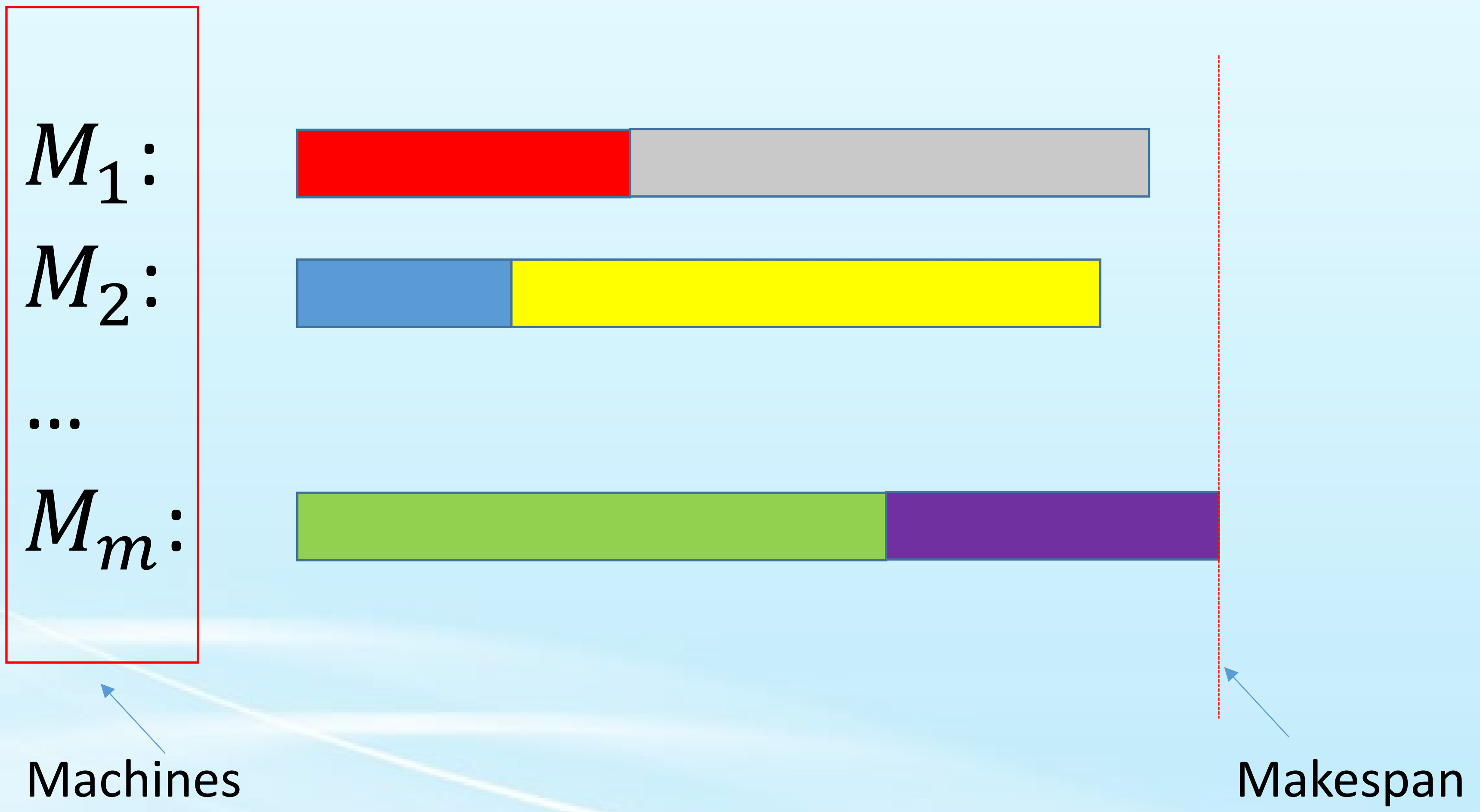
Lower Bounds for the Classical Scheduling Problem

Guochuan Zhang

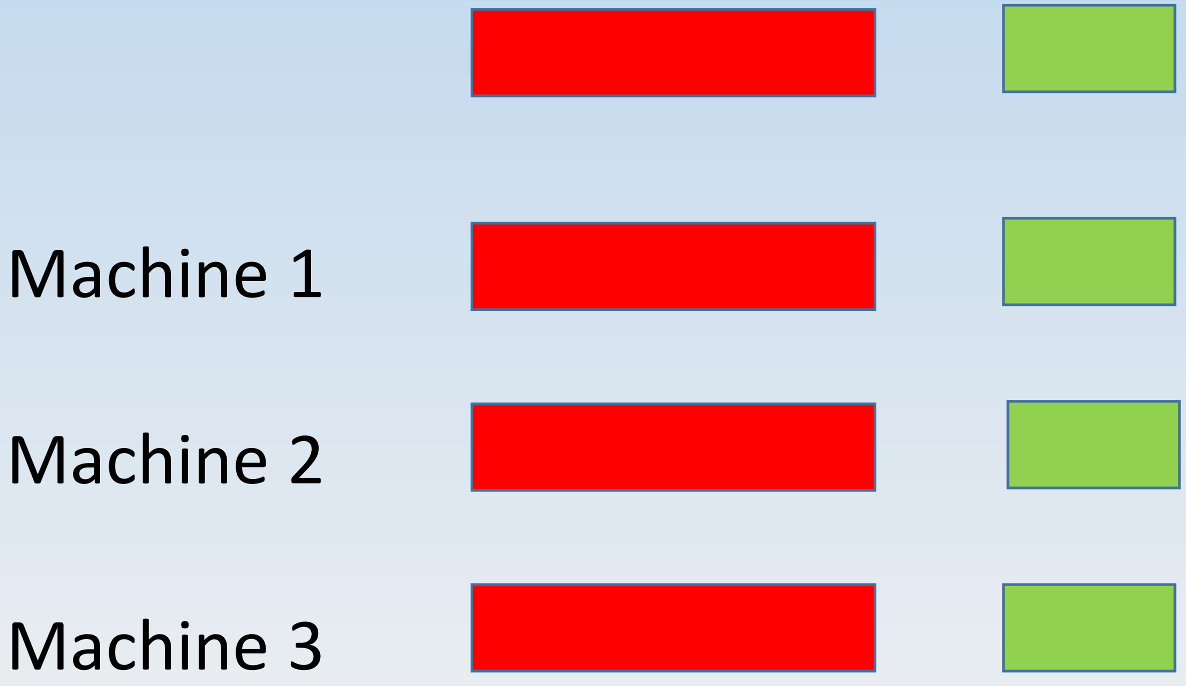
Joint with Lin Chen (TU Berlin), Klaus Jansen (Kiel), and Deshi Ye (Zhejiang)

April 1,2014 ,Aussois

The classical scheduling problem



Identical machines P//Cmax



Related machines Q//Cmax

Machine 1 (speed=1/2)

Machine 2 (speed=1)

Machine 3 (speed=2)

Unrelated machines R//Cmax

Known results

Approximation results

- $LPT\ (Longest\ Proocessing\ Time)$: $\frac{4}{3} - \frac{1}{3m}$ for P//Cmax
- $PTAS$ for Q//Cmax
- $FPTAS$ for Rm//Cmax
- $\left(2 - \frac{1}{m}\right)$ -approximation for R//Cmax
- $(\frac{3}{2} - \epsilon)$- approximation for R//Cmax implies P=NP

Two topics

- Improving running time for approximation schemes

- Improving performance rations for R//Cmax

Lower bounds on running time

Approximation schemes for $P||C_{max}$

- $(n/\epsilon)^{O(1/\epsilon^2)}$ (Hochbaum and Shmoys, JACM 1987)
- $f(1/\epsilon) + O(n)$ doubly exponential in $1/\epsilon$ (Alon et al., SODA 1998)
- $2^{O(1/\epsilon^2 \log^3 1/\epsilon)} + n^{O(1)}$ (Jansen, SIAM J. Discrete Mathematics 2010)

Approximation schemes for $Pm||C_{max}$

- $O(nm(nm/\epsilon)^{m-1})$ (Horowitz and Sahni, JACM 1976)
- $(n\ +\ 1)^{m/\epsilon}\,poly(|I|)$ (Lenstra, Shmoys, Tardos, Math. Prog 1990)
- $n(m/\epsilon)^{O(m)}$ (Jansen and Porkolab, MOR 2001)
- $O(n)\ +\ (1/\epsilon)^{O(m)}$ when $1/\epsilon > m$ (Jansen and Mastrolilli, 2010)

Approximation schemes for $Rm||C_{max}$

- $O(nm(nm/\epsilon)^{m-1})$ (Horowitz and Sahni, JACM 1976)
- $(n\ +\ 1)^{m/\epsilon}\ poly(|I|)$ (Lenstra, Shmoys, Tardos, Math. Prog 1990)
- $n(m/\epsilon)^{O(m)}$ (Jansen and Porkolab, MOR 2001)
- $O(n)\ +\ (1/\epsilon)^{O(m)}$ when $1/\epsilon > m$ (Jansen and Mastrolilli, 2010)

Our contribution

Algorithms	Upper bounds	Lower bounds
Approximation schemes	$2^{O(1/\epsilon^2 \log^3 1/\epsilon)} + \mathrm{n}^{O(1)}$	$2^{O((1/\epsilon)^{1-\delta})} + \mathrm{n}^{O(1)}$
Approximation schemes	$(1/\epsilon)^{O(m)} + O(n)$	$(1/\epsilon)^{O(m^{1-\delta})} + n^{O(1)}$
Exact algorithm	$2^{O(\sqrt{m\|I\| \log \quad m})}$	$2^{O(m^{1/2-\delta}\sqrt{\|I\|})}$
Exact algorithm	$2^{O(n)}$	$2^{O(n^{1-\delta})}$

Based on Exponential Time Hypothesis (ETH), assuming that 3SAT could not be solved in 2^{sn} time for some s>0 (Impagliazzo, Paturi, Zane, 2001)

A small gap for $P||C_{max}$

- $2^{O(1/\epsilon^2 \log^3 1/\epsilon)} + \mathrm{n}^{O(1)}$ $vs.$ $2^{O((1/\epsilon)^{1-\delta})} + \mathrm{n}^{O(1)}$

There exists a $2^{O(1/\epsilon \log^2 1/\epsilon)} + \mathrm{n}^{O(1)}$ - time algorithm,

(1) if every machine can accommodate at most a constant number of jobs;

(2) if a certain conjecture holds (Jansen and Robenek, WAOA 2011)

Reduction for $P||C_{max}$

- 2^{sn} for 3SAT → $2^{O((1/\epsilon)^{1-\delta})}$ for Scheduling

Goal

- 3SAT instance of $O(n)$ variables and clauses

→ A scheduling instance of makespan $L = O\left(n^k\right)$ ⇒ $\boldsymbol{L = O(n^{1+\delta})}$

A lower bound of $2^{O((1/\epsilon)^{1/k-\delta})}$ by taking $1/\epsilon = L + 1 = O(n^k)$
(A $(1+\epsilon)$-alg. is able to distinguish the makespan of L and L+1)

$k = 16$ in the traditional reduction

Reduction for $Pm||C_{max}$

- 2^{sn} for 3SAT → $(1/\epsilon)^{O(m^{1-\delta})}$ for Scheduling

Goal

- 3SAT instance of $O(n)$ variables and clauses

→ A scheduling instance of makespan $\mathrm{L} = 2^{O(\frac{n}{m}\log^2 m)}$

Taking $1/\epsilon = L + 1$, A $(1+\epsilon)$-algorithm is able to distinguish the makespan of L and L+1, running in $2^{\delta_m n}$ time with $\delta_m \to 0$ when $m \to \infty$

Open problems

- Close the gap between $2^{O(1/\epsilon^2 \log^3 1/\epsilon)} + \mathrm{n}^{O(1)}$ *and* $2^{O((1/\epsilon)^{1-\delta})} + \mathrm{n}^{O(1)}$ for $P||C_{max}$

- Influence of additional constraints: $P|r_j|C_{max}$

- Lower bounds on running times of approximation schemes for other classical objectives: $P||\sum w_j C_j$, $Q||\sum w_j C_j$, $P||\sum C_i^p$

Lower bound
for lower rank scheduling

Matrix of job processing times

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1,m} \\ p_{21} & p_{22} & \cdots & p_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n,1} & p_{n,2} & \cdots & p_{n,m} \end{bmatrix}$$

More precisely, the rank d comes from:

$$\begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1,m} \\ p_{21} & p_{22} & \cdots & p_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n,1} & p_{n,2} & \cdots & p_{n,m} \end{bmatrix} = \underbrace{\begin{bmatrix} * & * \\ * & * \\ \vdots & \vdots \\ * & * \end{bmatrix}}_{n * d} \underbrace{\begin{bmatrix} * & * & \cdots & * \\ * & * & \cdots & * \end{bmatrix}}_{d * m}$$

Rank of scheduling

- Identical machines: Rank=1
- Related machines: Rank=1
- Unrelated machines: Rank could be arbitrary

The hardness of the problem and the rank of matrix?

Bhaskara, Krishnaswamy, Talwar, Wieder, SODA 2013

Rank of Matrix	Hardness
1	PTAS (Hauchbaum and Shmoys)
2	QPTAS
3	open
4	APX-hard
≥7	(3/2-ϵ)-inapproximate

New lower bounds

Rank of Matrix	Hardness
3	APX-hard
≥4	(3/2-∊)-inapproximate

Rank 4 scheduling

Reduction from 3DM

- ***Element sets***: $W \cup X \cup Y, |\mathrm{W}| = |\mathrm{X}| = |\mathrm{Y}|$
- ***Match set***: $T \subset W \times X \times Y$
- ***Perfect matching***: $T' \subset T, every\ element\ appears\ once$

NP-hard to decide whether there exists a perfect matching

Rank 7 scheduling

Bhaskara, et al. 2013

- One machine for every match:
  $(w_i, x_j, y_k) \to (N^i, N^{-i}, N^j, N^{-j}, N^k, N^{-k}, 1)$

- Real jobs:
  $x_j \to (\ 0,\quad 0,\quad \epsilon N^{-j},\quad \epsilon N^j,\quad 0,\quad 0,\quad 1)$
  $y_k \to (\ 0,\quad 0,\quad 0,\quad 0,\quad \epsilon N^{-k}\quad \epsilon N^k,\ 1)$

- Dummy jobs: $d(w_i) - 1$ copies
  $w_i \to (\epsilon N^{-i}, \epsilon N^i,\quad 0,\quad 0,\quad 0,\quad 0,\quad 2)$

Rank 7 scheduling

Bhaskara, et al. 2013

- The processing time of each real job is at least 1
- The processing time of a dummy job is at least 2

  -- Ensured by the 7-th coordinate

- A w-job has to be scheduled on a machine (match) containing element w

  -- Ensured by the first 2 coordinates

Rank 7 scheduling

Bhaskara, et al. 2013

A schedule of length $2 + O(\epsilon)$

$(w_1,*,*)$ $w_1 - dummy$

$(w_1,*,*)$ $w_1 - dummy$

$\dots$

(w_1, x_j, y_k) $x_j - job, y_k - job$

$(w_2,*,*)$ $w_2 - dummy$

$\dots$

Rank 4 scheduling

$$LRS(7) \leftarrow (*, \quad *, \quad *, \quad *, \quad *, \quad *, \quad *)$$

$w - coordinates$ $\quad$ $x - coordinates$ $\quad$ $y - coordinates$

Let w, x, y share coordinates

However, in the 3DM problem, for a match (w_i, x_j, y_k), the indices (i,j,k) could be arbitrary

A variant of 3DM

Reduction from a special 3DM (still NP-complete)

- Element sets: $W = \{w_i, \bar{w}_i | i \in [3n]\}, X = \{s_i, a_i | \ i \in [3n]\}, Y = \{s'_i, b_i | \ i \in [3n]\}$
- Match set: $T_1 \subset \left\{ \left(w_i, s_j, s'_j\right), \left(\bar{w}_i, s_j, s'_j\right) \middle| w_i \in W, s_j \in X, s'_j \in Y \right\}$
  $T_2 = \{(w_i, a_i, b_i), (\bar{w}_i, a_i, b_{\zeta(i)}) | \ i \in [3n]\}$

$\zeta: \zeta(3k+1) = 3k+2, \zeta(3k+2) = 3k+3, \zeta(3k+3) = 3k+1$

Tied indices

(w_i, s_j, s'_j), $(\bar{w}_i, s_j, s'_j)$, (w_i, a_i, b_i), $(\bar{w}_i, a_i, b_{\zeta(i)})$

Indices are tied

Overall structure

Machines

- (w_i, s_j, s_j'): $(N^i, N^{-i}, N^{j+N}, N^{-j-N})$
- $(\bar{w}_i, s_j, s_j')$: $(N^{-i}, N^i, N^{j+N}, N^{-j-N})$
- $(w_i, a_i, *)$: $(N^i, N^{-i}, N^{-i}, *)$
- $(\bar{w}_i, a_i, *)$: $(N^{-i}, N^i, N^{-i}, *)$

Jobs

- w_i: $(N^{-i}, N^i, 0, 0)$
- $\bar{w}_i$: $(N^i, N^{-i}, 0, 0)$
- a_i: $(N^{-i}, N^{-i}, N^i, 0)/2$
- $s_j(s_j')$: $(0, 0, N^{-j-N}, N^{j+N})/2$

Overall structure

Machines

- (w_{3i}, a_{3i}, b_{3i}): $\left(N^{3i}, \quad N^{-3i}, \quad N^{-3i}, \quad N^{-3i-1}\right)$
- $(w_{3i+1}, a_{3i+1}, b_{3i+1})$: $\left(N^{3i+1}, \quad N^{-3i-1}, \quad N^{-3i-1}, \quad N^{-3i}\right)$
- $(w_{3i+2}, a_{3i+2}, b_{3i+2})$: $\left(N^{3i+2}, \quad N^{-3i-2}, \quad N^{-3i-2}, \quad \frac{1}{\epsilon}N^{-3i-1}\right)$
- $(\overline{w}_{3i}, a_{3i}, b_{3i+1})$: $\left(N^{-3i}, \quad N^{3i}, \quad \epsilon N^{-3i}, \quad N^{-3i}\right)$
- $(\overline{w}_{3i+1}, a_{3i+1}, b_{3i+2})$: $\left(N^{-3i-1}, \quad N^{3i+1}, \quad N^{-3i-1}, \quad \frac{1}{\epsilon}N^{-3i-1}\right)$
- $(\overline{w}_{3i+2}, a_{3i+2}, b_{3i})$: $\left(N^{-3i-2}, \quad N^{3i+2}, \quad \epsilon N^{-3i-2}, \quad N^{-3i-1}\right)$

Jobs

- b_{3i}: $\left(\epsilon N^{-3i}, \quad \epsilon N^{-3i-2}, \quad N^{3i}, \quad N^{3i+1}\right)/2$
- b_{3i+1}: $\left(\epsilon N^{-3i-1}, \quad \frac{1}{\epsilon}N^{-3i-1}, \quad N^{3i}, \quad N^{3i}\right)/2$
- b_{3i+2}: $\left(\epsilon N^{-3i-2}, \quad \epsilon N^{-3i-1}, \quad N^{3i+1}, \quad \epsilon N^{3i+1}\right)/2$

Gap

- If the instance of the special 3DM has a perfect matching, then there exists a schedule with a makespan of $2+O(\epsilon)$

- If the scheduling instance has a feasible schedule of makespan smaller than 3, then the 3DM has a perfect matching

- This gives a lower bound of $3/2-\epsilon$

Open problems

. Approximation algorithms for scheduling problem of a constant rank.
. Is rank 2 scheduling harder than rank 1 (in terms of schemes)?
. Is rank 4 scheduling harder than rank 3 (in terms of approx ratios)?

Thank You!