A Graphical Algorithm for Solving an Investment Optimization Problem

Evgeny R. Gafarov — Institute of Control Sciences of the Russian Academy of Sciences

Alexandre Dolgui — Ecole Nationale Superieure des Mines

Alexander A. Lazarev — Institute of Control Sciences of the Russian Academy of Sciences

Frank Werner — Otto-von-Guericke Universität Magdeburg

Outline

- Problem formulation;
- Dynamic programming and graphical algorithms;
- FPTAS for 6 scheduling problems.

Investment Problem

n investment projects

A – investment budget (A arbitrarily given from some interval $[A', A'']$)

$f_j(t)$ -- profit function of project j

The objective is to define an integer amount t_j in $[0,A]$ for each project to maximize the total profit.

$\sum t_j <= A$

$\sum$ fj(tj) max
(t_j is integer)

S. Kameshwaran and Y. Narahari, Nonconvex Piecewise Linear Knapsack Problems, European Journal of Operational Research, 192, 2009, 56 - 68.

$$O\left(\frac{(\sum k_j)^3}{\epsilon}\right)$$

A special case of this problem is similar to the well-known bounded knapsack problem:

$$\begin{aligned} &\text{maximize} \; \textstyle\sum_{j:=1}^{n} p_j x_j \\ &s.t. \qquad \textstyle\sum_{j:=1}^{n} w_j x_j \leq A, \\ &\qquad\qquad x_j \in [0, b_j],\; x_j \in Z,\; j = 1, 2, \ldots, n, \end{aligned} \tag{1}$$

for which a dynamic programming algorithm (DPA) of time complexity $O(nA)$ is known [3].

3. H. Kellerer, U. Pferschy and D. Pisinger, Knapsack Problems, Springer-Verlag, Berlin, 2004.

The following problem is also similar to the problem under consideration:

$$\begin{aligned} &\text{minimize } \sum_{j:=1}^{n} f_j(x_j) \\ &s.t. \quad \sum_{j:=1}^{n} x_j \geq A, \\ &\qquad x_j \in [0, A],\ x_j \in Z,\ j = 1, 2, \dots, n, \end{aligned} \tag{2}$$

where $f_j(x_j)$ are piecewise linear as well. For this problem, a DPA with a running time of $O(\sum k_j A)$ [4] and a fully polynomial-time approximation scheme (FPTAS) with a running time of $O((\sum k_j)^3/\varepsilon)$ [5] are known.

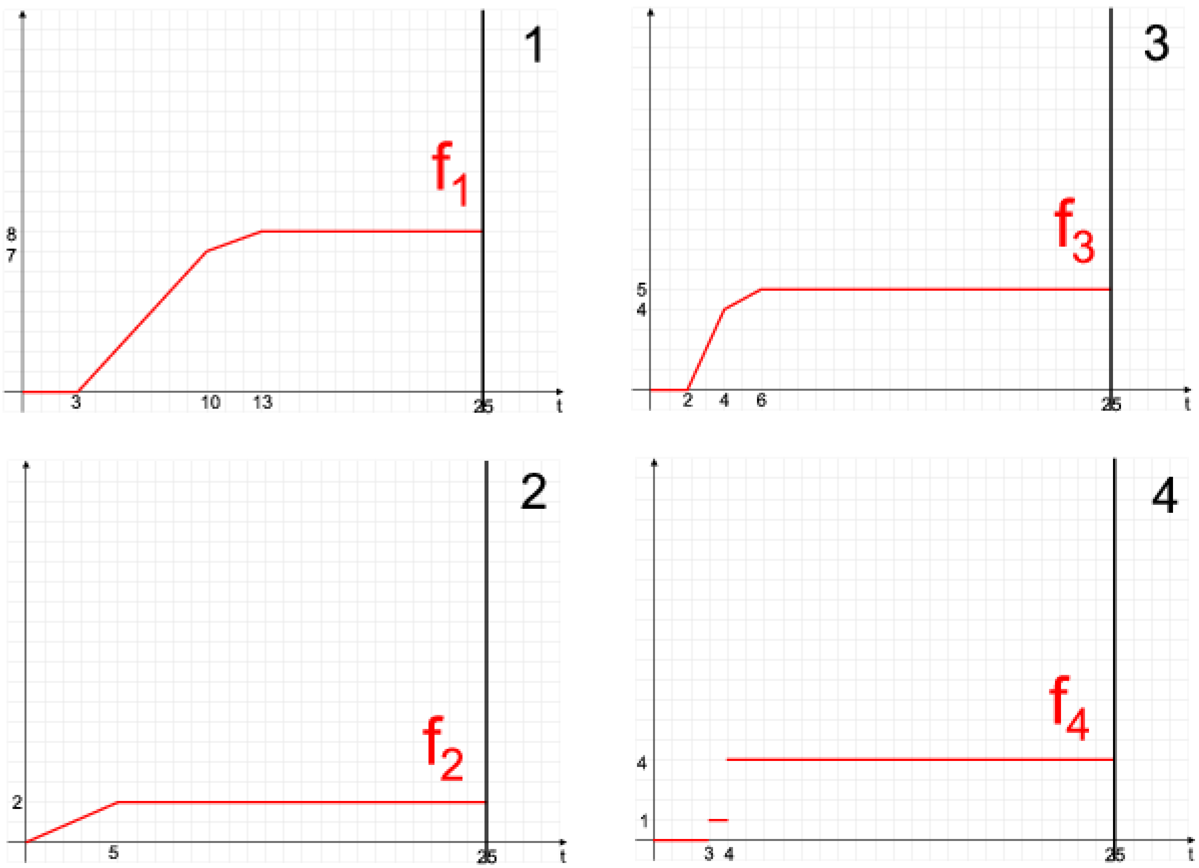
4. D.X. Shaw and A. P. M. Wagelmans, An Algorithm for Single-Item Capacitated Economic Lot Sizing with Piecewise Linear Production Costs and General Holding Costs, Management Science, Vol. 44, No. 6, 1998, 831-838.
5. S. Kameshwaran and Y. Narahari, Nonconvex Piecewise Linear Knapsack Problems, European Journal of Operational Research, 192, 2009, 56- 68.

In this paper, we deal with piecewise linear functions $f_j(x)$. Suppose that the interval $[0, A]$ can be written as

$$[0, A] = [t_j^0, t_j^1] \bigcup (t_j^1, t_j^2] \bigcup \ldots \bigcup (t_j^{k-1}, t_j^k] \bigcup \ldots \bigcup (t_j^{k_j-1}, t_j^{k_j}]$$

such that the profit function has the form $f_j(x) = b_j^k + u_j^k(x - t_j^{k-1})$, if $x \in (t_j^{k-1}, t_j^k]$, where k is the number of the interval, b_k^j is the value of the function at the beginning of the interval, and u_j^k is the slope of the function. Without loss of generality, assume that $b_j^1 \le b_j^2 \le \ldots \le b_j^{k_j}$ and $t_j^k \in Z,\ j \in N,\ k = 1, 2, \ldots, k_j$, and that $t_j^{k_j} = A,\ j = 1, 2, \ldots, n$.

Investment Problem



K	1	2	3	4
interval K	$[0, 3)$	$[3, 10)$	$[10, 13)$	$[13, 25]$
b_1^K	0	0	7	8
u_1^K	0	1	$\frac{1}{3}$	0

Graphical Algorithms for the Investment Problem

Running time of the classical dynamic programming algorithm: $O(nA^2)$.
Running time of the best known dynamic programming algorithm: $O(\sum k_j A)$.

$$F_j(T) = \max_{t=0,1,\ldots,T} \{f_j(t) + F_{j-1}(T-t)\},\ T = A, A-1, \ldots, 1,$$

In the graphical algorithm, the functions $f_j(t)$ and the Bellman functions (value function) $F_j(t)$ are saved in a tabular form:

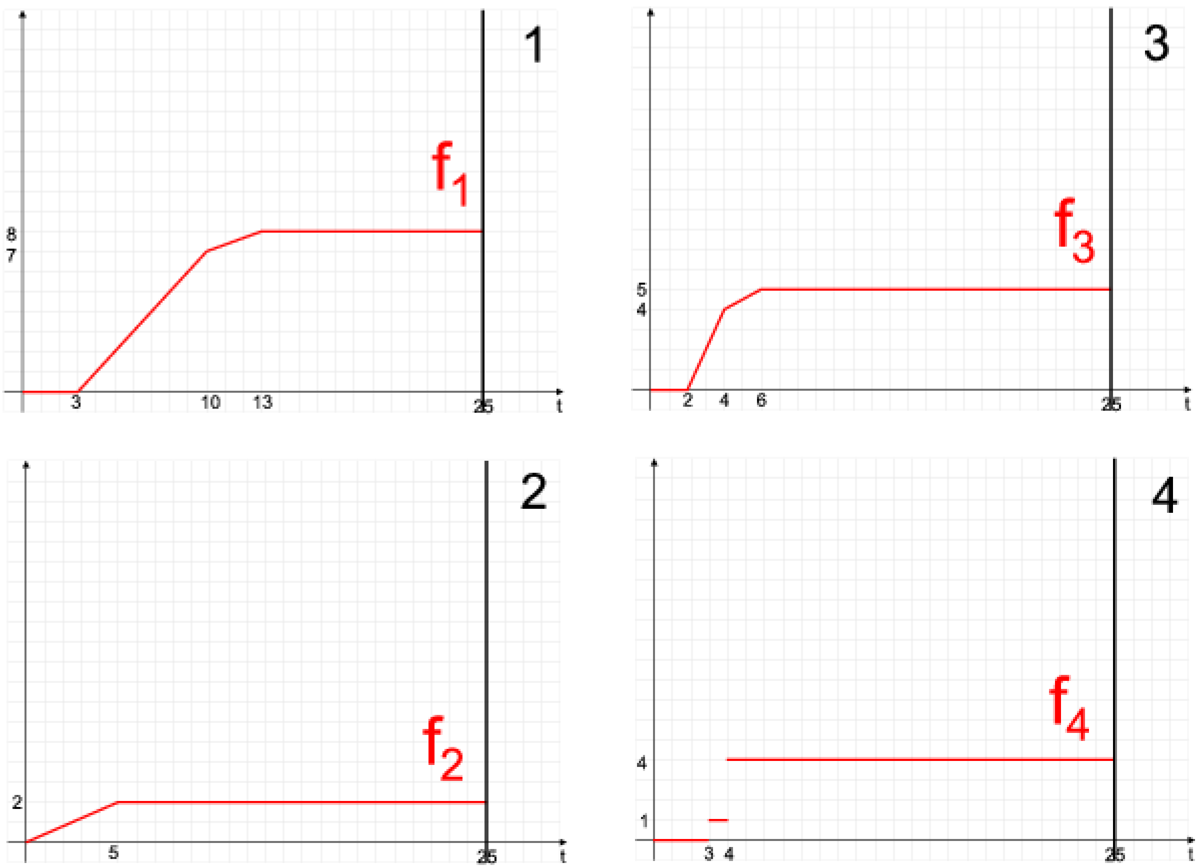
K	1	2	$\ldots$	k_j
interval K	$[t_j^1, t_j^2)$	$[t_j^2, t_j^3)$	$\ldots$	$[t_j^{k_j}, A)$
b_j^K	b_j^1	b_j^2	$\ldots$	$b_j^{k_j}$
u_j^K	u_j^1	u_j^2	$\ldots$	$u_j^{k_j}$

Running time of the 1st version of the graphical algorithm: $O(nk_{max}A \log(k_{max}A))$

Running time of the 2nd version of the graphical algorithm: $O(\sum k_j A)$

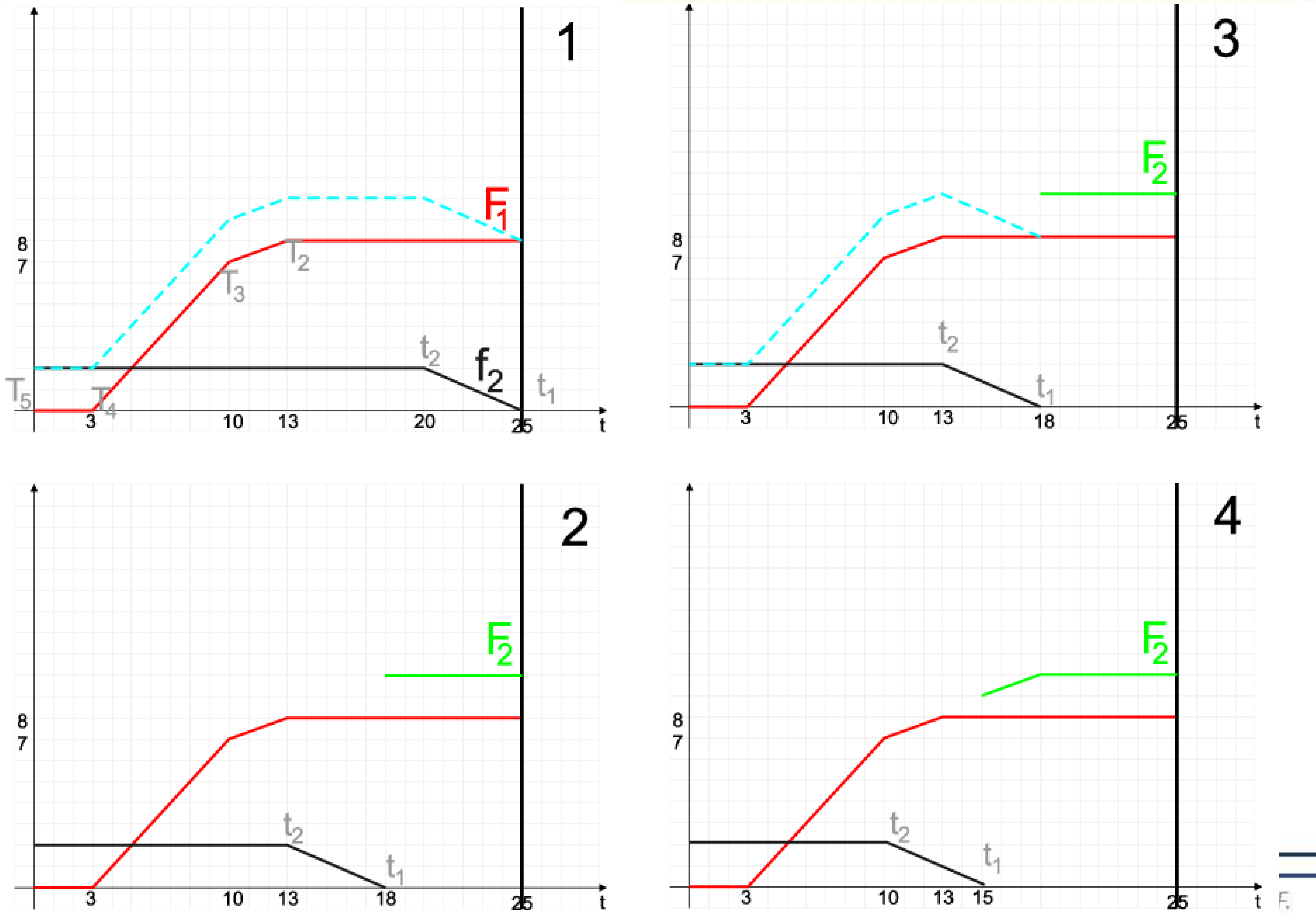
Running time of the FPTAS based on the graphical algorithm: $O(n(\log\log n)\sum k/\varepsilon)$

Graphical algorithm for Investment problem



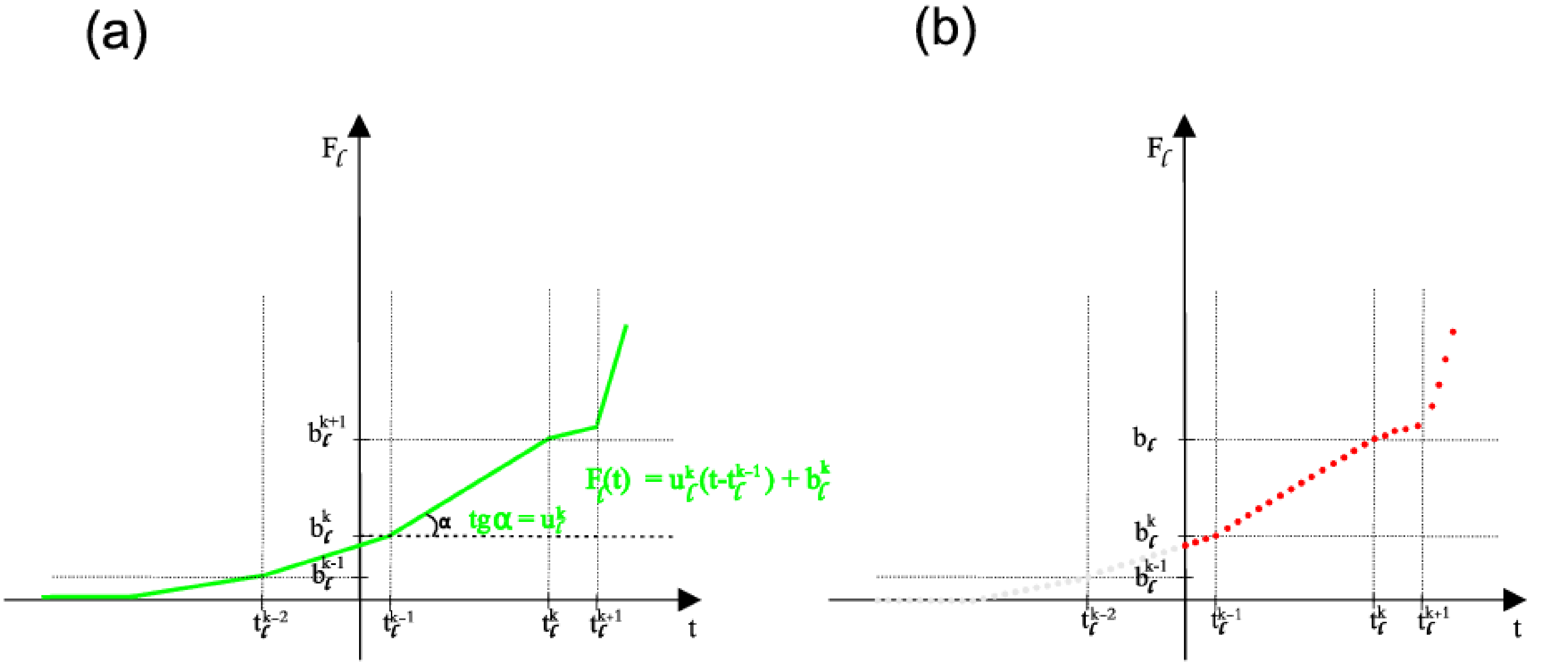
K	1	2	3	4
interval K	$[0, 3)$	$[3, 10)$	$[10, 13)$	$[13, 25]$
b_1^K	0	0	7	8
u_1^K	0	1	$\frac{1}{3}$	0

A Graphical Algorithm for Solving an Investment Optimization Problem



In this paper, we present an alternative solution algorithm with a running time of $O(\sum k_j A)$ and an FPTAS based on this solution algorithm with a running time of $O(\sum k_j n \log\log n/\varepsilon)$.

FPTAS for 6 scheduling problems



FPTAS based on the Graphical Algorithm

k	1	2	...	m_j+1	m_j+2
interval k	$(-\infty, t_j^1]$	$(t_j^1, t_j^2]$	...	$(t_j^{m_j}, t_j^{m_j+1}]$	$(t_j^{m_j+1}, +\infty)$
b_j^k	0	b_j^2	...	$b_j^{m_j+1}$	$+\infty$
u_j^k	0	u_j^2	...	$u_j^{m_j+1}$	0
π_j^k	π_j^1	π_j^2	...	$\pi_j^{m_j+1}$	$(1,2,\ldots,j)$

In the table, $0<b_l^1<b_l^2<\ldots$ since function $F(t)$ is monotonic with t being the starting time.

The running time of the graphical algorithm is $O(n \min\{UB,d\})$ for each straddling job x.

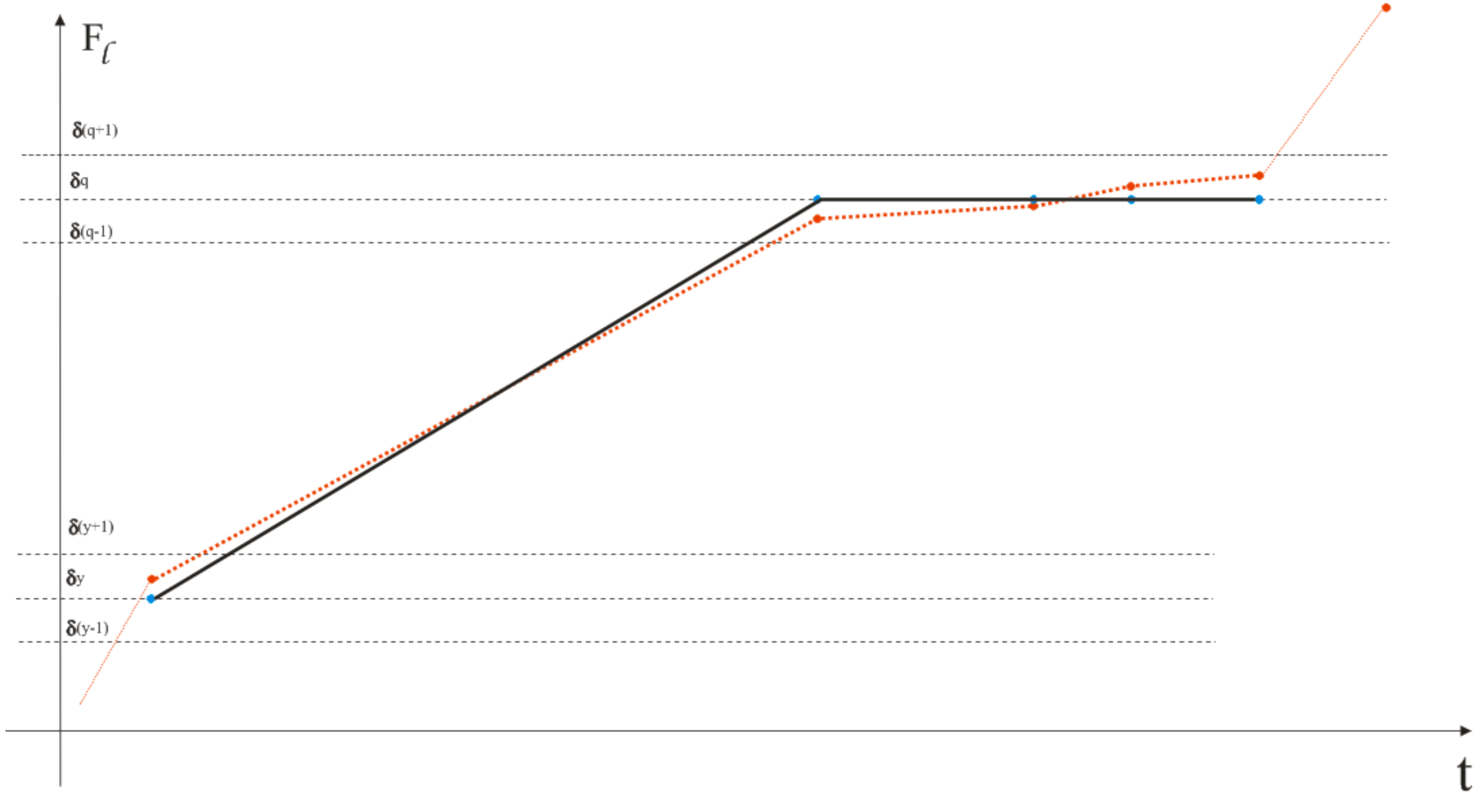
To reduce the running time, we can round (approximate) the values $b_l^k<UB$ to get a polynomial number of different values b_l^k

Let $\delta = \frac{\varepsilon UB}{2n}$. Round b_l^k up or down to the nearest multiple of δ

The idea of the FPTAS is as follows. Let $\delta = \frac{\varepsilon LB}{n}$. To reduce the time complexity of the GrA, we have to diminish the number of columns $|F_j.B|$ considered, which corresponds to the number of different objective function values $b \in F_j.B, b \leq UB$. If we do not consider the original values $b \in F_j.B$ but the values $\overline{b}$ which are rounded up or down to the nearest multiple of δ values b, there are no more than $\frac{UB}{\delta} = \frac{n^2}{\varepsilon}$ different values $\overline{b}$. Then we will be able to approximate the function $F_j(t)$ into a similar function with no more than $2\frac{n^2}{\varepsilon}$ break points. Furthermore, for such a modified table representing a function $\overline{F}_j(t)$, we will have

$$|F_j(t) - \overline{F_j}(t)| < \delta \leq \frac{\varepsilon F(\pi^*)}{n}.$$

FPTAS based on the Graphical Algorithm



no more than $\frac{UB}{\delta} = \frac{2n}{\varepsilon}$ different values $\overline{b_l^k}$

no more than $4\frac{n}{\varepsilon}$ columns

cumulative error will be no more than $n\delta \leq \varepsilon F(\pi^*)$

The running time of the FPTAS is $O(\frac{n^3}{\varepsilon})$

FPTAS for 6 scheduling problems

Problem	Time complexity of the GrA	Time complexity of the FP-TAS	Time complexity of the classical DPA
$1\|\|\sum w_jU_j$	$O(\min\{2^n, n \cdot \min\{d_{max}, F_{opt}\}\})$ [5]	-	$O(nd_{max})$
$1\|d_j = d'_j + A\|\sum U_j$	$O(n^2)$ [5] (GrA)	-	$O(n \sum p_j)$
$1\|\|\sum GT_j$	$O(\min\{2^n, n \cdot \{d_{max}, nF^*\}\})$	$O(n^2 \log\log n + \frac{n^2}{\varepsilon})$	$O(nd_{max})$
$1\|\|\sum T_j$ special case $B-1$	$O(\min\{2^n, n \cdot \min\{d_{max}, F^*\}\})$	$O(n^2/\varepsilon)$	$O(nd_{max})$
$1\|\|\sum T_j$ special case $B-1G$	$O(\min\{n^2 \cdot \min\{d_{max}, F^*\}\})$	$O(n^3/\varepsilon)$	$O(n^2d_{max})$
$1\|d_j = d\|\sum w_jT_j$	$O(\min\{n^2 \cdot \min\{d, F^*\}\})$	$O(n^3/\varepsilon)$	$O(n^2d_{max})$
$1(no\text{-}idle)\|\|\max \sum w_jT_j$	$O(\min\{2^n, n \cdot \min\{d_{max}, nF^*, \sum w_j\}\})$ [5]	$O(n^2 \log\log n + \frac{n^2}{\varepsilon})$	$O(nd_{max})$
$1(no\text{-}idle)\|\|\max \sum T_j$	$O(n^2)$ [4] (GrA)	-	$O(nd_{max})$

Thanks for attention

Gafarov Evgeny, Dolgui Alexandre, Lazarev Alexander, Werner Frank

Ecole Nationale Supérieure des Mines SAINT-ETIENNE

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG