

Optimal Composition Ordering Problems for Piecewise Linear Functions*

Yasushi Kawase¹, Kazuhisa Makino², and Kento Seimi³

1 School of Engineering, Tokyo Institute of Technology, Japan
kawase.y.ab@m.titech.ac.jp

2 Research Institute for Mathematical Sciences, Kyoto University, Japan
makino@kurims.kyoto-u.ac.jp

3 Tokyo, Japan
kento.seimi@gmail.com

Abstract

In this paper, we introduce maximum composition ordering problems. The input is n real functions $f_1, \dots, f_n : \mathbb{R} \rightarrow \mathbb{R}$ and a constant $c \in \mathbb{R}$. We consider two settings: total and partial compositions. The maximum total composition ordering problem is to compute a permutation $\sigma : [n] \rightarrow [n]$ which maximizes $f_{\sigma(n)} \circ f_{\sigma(n-1)} \circ \dots \circ f_{\sigma(1)}(c)$, where $[n] = \{1, \dots, n\}$. The maximum partial composition ordering problem is to compute a permutation $\sigma : [n] \rightarrow [n]$ and a nonnegative integer k ($0 \leq k \leq n$) which maximize $f_{\sigma(k)} \circ f_{\sigma(k-1)} \circ \dots \circ f_{\sigma(1)}(c)$.

We propose $O(n \log n)$ time algorithms for the maximum total and partial composition ordering problems for monotone linear functions f_i , which generalize linear deterioration and shortening models for the time-dependent scheduling problem. We also show that the maximum partial composition ordering problem can be solved in polynomial time if f_i is of the form $\max\{a_i x + b_i, c_i\}$ for some constants $a_i (\geq 0)$, b_i and c_i . As a corollary, we show that the two-valued free-order secretary problem can be solved in polynomial time. We finally prove that there exists no constant-factor approximation algorithm for the problems, even if f_i 's are monotone, piecewise linear functions with at most two pieces, unless $P=NP$.

1998 ACM Subject Classification G.2.1 Combinatorial algorithms

Keywords and phrases function composition, time-dependent scheduling

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2016.42

1 Introduction

In this paper, we introduce *optimal composition ordering problems* and mainly study their time complexity. The input of the problems is n real functions $f_1, \dots, f_n : \mathbb{R} \rightarrow \mathbb{R}$ and a constant $c \in \mathbb{R}$. In this paper, we assume that the input functions are piecewise linear, and the input length of a piecewise linear function is the sum of the sizes of junctions and coefficients of linear functions. We consider two settings: *total* and *partial* compositions. The maximum total composition ordering problem is to compute a permutation $\sigma : [n] \rightarrow [n]$ that maximizes $f_{\sigma(n)} \circ f_{\sigma(n-1)} \circ \dots \circ f_{\sigma(1)}(c)$, where $[n] = \{1, \dots, n\}$. The maximum partial composition ordering problem is to compute a permutation $\sigma : [n] \rightarrow [n]$ and a nonnegative integer k ($0 \leq k \leq n$) that maximize $f_{\sigma(k)} \circ f_{\sigma(k-1)} \circ \dots \circ f_{\sigma(1)}(c)$. For example, if the input consists of $f_1(x) = 2x - 6$, $f_2(x) = \frac{1}{2}x + 2$, $f_3(x) = x + 2$, and $c = 2$, then the ordering σ such

* The first author is supported by JSPS KAKENHI Grant Number 26887014 and 16K16005. The second author is supported by JSPS KAKENHI Grant.



that $\sigma(1) = 2$, $\sigma(2) = 3$, and $\sigma(3) = 1$ is optimal for the maximum total composition ordering problem. In fact, $f_1 \circ f_3 \circ f_2(c) = f_1(f_3(f_2(c))) = f_1(f_3(c/2 + 2)) = f_1(c/2 + 4) = c + 2 = 4$ provides the optimal value of the problem. The ordering σ above and $k = 2$ is optimal for the maximum partial composition ordering problem, where $f_3 \circ f_2(c) = 5$. We also consider the maximum exact k -composition ordering problem, which is a problem to compute a permutation $\sigma : [n] \rightarrow [n]$ that maximizes $f_{\sigma(k)} \circ f_{\sigma(k-1)} \circ \dots \circ f_{\sigma(1)}(c)$ for given n functions $f_1, \dots, f_n : \mathbb{R} \rightarrow \mathbb{R}$, a constant $c \in \mathbb{R}$, and a nonnegative integer k ($0 \leq k \leq n$). We remark that the minimization versions are reducible to the maximization ones.

As we will see in this paper, the optimal composition ordering problems are natural and fundamental in many fields such as artificial intelligence, computer science, and operations research. However, to the best of the authors' knowledge, no one explicitly studies the problems from the algorithmic point of view. We below describe single machine time-dependent scheduling problems and the free-order secretary problem, which can be formulated as the optimal composition ordering problems.

Time-dependent scheduling

Consider machine scheduling problems with time-dependent processing times, called *time-dependent scheduling problems* [6, 12].

Let J_i ($i = 1, \dots, n$) denote a job with a ready time $r_i \in \mathbb{R}$, a deadline $d_i \in \mathbb{R}$, and a processing time $p_i : \mathbb{R} \rightarrow \mathbb{R}$, where $r_i \leq d_i$ is assumed. Different from the classical setting, the processing time p_i is *not* constant, but depends on the *starting* time of job J_i . The model has been studied to deal with learning and deteriorating effects, for example [13, 14, 15, 19, 20]. Here each p_i is assumed to satisfy $p_i(t) \leq s + p_i(t + s)$ for any t and $s \geq 0$, since we should be able to finish processing job J_i earlier if it starts earlier. Among time-dependent settings, we consider the single machine scheduling problem to minimize the makespan, where the input is the start time t_0 ($= 0$) and a set of J_i ($i = 1, \dots, n$) above. The makespan denotes the time when all the jobs have finished processing, and we assume that the machine can handle only one job at a time and preemption is not allowed. We show that the problem can be viewed as the minimum total composition ordering problem.

Define a function f_i by

$$f_i(t) = \begin{cases} r_i + p_i(r_i) & (t \leq r_i), \\ t + p_i(t) & (r_i < t \leq d_i - p_i(t)), \\ \infty & (t > d_i - p_i(t)). \end{cases}$$

Then the problem can be reduced to the minimum total composition ordering problem for $(f_i)_{i \in [n]}$ and $c = t_0$.

A number of restrictions on the processing time $p_i(t)$ has been studied in the literature (e.g., [3, 5, 17]).

In the *linear deterioration* model, the processing time p_i is restricted to be a monotone increasing linear function that satisfies $p_i(t) = a_i t + b_i$ for two positive constants a_i and b_i . Here a_i and b_i are respectively called the *deterioration rate* and the *basic processing time* of job J_i . Gawiejnowicz and Pankowska [13], Gupta and Gupta [14], Tanaev *et al.* [19], and Wajs [20] obtained the result that the time-dependent scheduling problem of this model (without the ready time r_i nor the deadline d_i) is solvable in $O(n \log n)$ time by scheduling jobs in the nonincreasing order of the ratios b_i/a_i . As for the hardness results, it is known that the proportional deterioration model with ready time and deadline, the linear

deterioration model with ready time, and the linear deterioration model with deadlines are all NP-hard [4, 11].

Another model is called the *linear shortening* model introduced by Ho *et al.* [15]. In this model, the processing time p_i is restricted to be a monotone decreasing linear function that satisfies $p_i(t) = -a_i t + b_i$ with a_i and b_i such that $1 > a_i > 0$, $b_i > 0$. They showed that the time-dependent scheduling problem of this model can be solved in $O(n \log n)$ time by again scheduling jobs in the nonincreasing order of the ratios b_i/a_i .

Free-order secretary problem

The *free-order secretary problem* is another application of the optimal composition ordering problems, which is closely related to a branch of the problems such as the full-information secretary problem [9], knapsack and matroid secretary problems [1, 2, 18] and stochastic knapsack problems [7, 8]. Imagine that an administrator wants to hire the best secretary out of n applicants for a position. Each applicant i has a nonnegative independent random variable X_i as his ability for the secretary. Here X_1, \dots, X_n are not necessarily based on the same probability distribution, and assume that the administrator knows all the probability distributions of X_i 's before their interviews, where such information can be obtained by their curriculum vitae and/or results of some written examinations. The applicants are interviewed one-by-one, and the administrator can observe the value X_i during the interview of the applicant i . A decision on each applicant is to be made immediately after the interview. Once an applicant is rejected, he will never be hired. The interview process is finished if some applicant is chosen, where we assume that the last applicant is always chosen if he is interviewed since the administrator has to hire exactly one candidate. The objective is to find an optimal strategy for this interview process, i.e., to find an interview ordering together with the stopping rule that maximizes the expected value of the secretary hired.

Let $f_i(x) = \mathbf{E}[\max\{X_i, x\}]$. We now claim that our secretary problem can be represented by the maximum total composition ordering problem $((f_i)_{i \in [n]}, c = 0)$.

Let us first consider the best stopping rule for the interview to maximize the expected value for the secretary hired when the interview ordering is fixed in advance. Assume that the applicant i is interviewed in the i th place. Note that $\mathbf{E}[X_n] (= f_n(0))$ is the expected value under the condition that all the applicants except for the last one are rejected, since the last applicant is hired. Consider the situation that all the applicants except for the last two ones are rejected. Then it is a best stopping rule that the applicant $n - 1$ is hired if and only if $X_{n-1} \geq f_n(0)$ is satisfied (i.e., the applicant n is hired if and only if $X_{n-1} < f_n(0)$), where $f_{n-1} \circ f_n(0)$ is the expected value for the best stopping rule, under this situation. By applying backward induction, we have the following best stopping rule: we hire the applicant $i (< n)$ and stop the interview process, if $X_i \geq f_{i+1} \circ \dots \circ f_n(0)$ (otherwise, the next applicant is interviewed), and we hire the applicant n if no applicant $i (< n)$ is hired. It turns out that $f_1 \circ \dots \circ f_n(0)$ is the maximum expected value for the secretary hired, if the interview ordering is fixed such that the applicant i is interviewed in the i th place.

Therefore, the secretary problem (i.e., finding an interview ordering, together with a stopping rule) can be formulated as the maximum total composition ordering problem $((f_i)_{i \in [n]}, c = 0)$.

In addition, let us assume that X_i is an m -valued random variable that takes the value a_i^j with probability $p_i^j \geq 0$ ($j = 1, \dots, m$). Here we assume that $a_i^1 \geq \dots \geq a_i^m \geq 0$ and

$\sum_{j=1}^m p_i^j = 1$. Then we have

$$f_i(x) = \sum_{j=1}^m p_i^j \max\{a_i^j, x\} = \max_{l=0, \dots, m} \left\{ \sum_{j=1}^l p_i^j a_i^j + \sum_{j=l+1}^m p_i^j x \right\}.$$

Note that this f_i is a monotone convex piecewise linear function with at most $(m + 1)$ pieces.

Main results obtained in this paper

In this paper, we consider the computational issues for the optimal composition ordering problems, when all f_i 's are monotone and almost linear.

We first show that the problems become tractable if all f_i 's are monotone and linear, i.e., $f_i(x) = a_i x + b_i$ for $a_i \geq 0$.

► **Theorem 1.** *The maximum partial and total composition ordering problems for monotone nondecreasing linear functions are both solvable in $O(n \log n)$ time.*

Recall that the algorithm for the linear shortening model (resp., the linear deterioration model) for the time-dependent scheduling problem is easily generalized to the case when all a_i 's satisfy $a_i < 1$ (resp., $a_i > 1$). The best composition ordering is obtained as the nondecreasing order of the ratios b_i/a_i . This idea can be extended to the maximum partial composition ordering problem in the mixed case (i.e., some $a_i > 1$ and some $a_{i'} < 1$) of Theorem 1. However, we cannot extend it to the maximum total composition ordering problem. In fact, we do not know if there exists such a simple criterion on the maximum total composition ordering. We instead present an efficient algorithm that chooses the best ordering among linearly many candidates.

We also provide a dynamic-programming based polynomial-time algorithm for the exact k -composition setting.

► **Theorem 2.** *The maximum exact k -composition ordering problem for monotone nondecreasing linear functions is solvable in $O(k \cdot n^2)$ time.*

We next consider monotone, piecewise linear case. It can be directly shown from the time-dependent scheduling problem that the maximum total composition ordering problem is NP-hard, even if all f_i 's are monotone, concave, and piecewise linear functions with at most two pieces, i.e., $f_i(x) = \min\{a_i^1 x + b_i^1, a_i^2 x + b_i^2\}$ for some constants $a_i^1, a_i^2, b_i^1,$ and b_i^2 with $a_i^1, a_i^2 > 0$. It turns out that all the other cases become intractable, even if all f_i 's are monotone and consist of at most two pieces. Furthermore, the problems are inapproximable.

► **Theorem 3.**

- (i) *For any positive real number $\alpha (\leq 1)$, there exists no α -approximation algorithm for the maximum total (partial) composition ordering problem even if all f_i 's are monotone, **concave**, and piecewise linear functions with at most two pieces, unless $P=NP$.*
- (ii) *For any positive real number $\alpha (\leq 1)$, there exists no α -approximation algorithm for the maximum total (partial) composition ordering problem even if all f_i 's are monotone, **convex**, and piecewise linear functions with at most two pieces, unless $P=NP$.*

Here f_i can be represented by $f_i(x) = \max\{a_i^1 x + b_i^1, a_i^2 x + b_i^2\}$ for some constants $a_i^1, a_i^2, b_i^1,$ and b_i^2 with $a_i^1, a_i^2 > 0$ if f_i is a monotone, convex, and piecewise linear function with at most two pieces.

As for the positive side, if each f_i is a monotone, convex, and piecewise linear function with at most two pieces such that one of the pieces is constant, then we have the following

result, which implies that the two-valued free-order secretary problem can be solved in $O(n^2)$ time.

► **Theorem 4.** *Let $f_i(x) = \max\{a_i x + b_i, c_i\}$ for some constants $a_i (\geq 0)$, b_i and c_i . Then the maximum partial composition ordering problem is solvable in $O(n^2)$ time.*

Due to space limitation, we omit several proofs. They can be found in the full version of this paper [16].

2 Maximum Partial Composition Ordering Problem

In this section, we discuss tractable results for the maximum partial composition ordering problem for monotone and almost-linear functions. We deal with the problem as the maximum total composition ordering problem for functions \bar{f}_i ($i \in [n]$), where $\bar{f}_i(x) = \max\{f_i(x), x\}$. It is easy to see that the objective value of the maximum partial composition ordering problem $((f_i)_{i \in [n]}, c)$ is equal to the one of the maximum total composition ordering problem $((\bar{f}_i)_{i \in [n]}, c)$. Let us start with the maximum partial composition ordering problem for monotone linear functions $f_i(x) = a_i x + b_i$ ($a_i \geq 0$), i.e., the total composition ordering problem for $\bar{f}_i(x) = \max\{a_i x + b_i, x\}$ ($a_i \geq 0$).

The following binary relation \preceq plays an important role in the problem.

► **Definition 5.** For two functions $f, g : \mathbb{R} \rightarrow \mathbb{R}$, we write $f \preceq g$ (or $g \succeq f$) if $f \circ g(x) \leq g \circ f(x)$ for any $x \in \mathbb{R}$, $f \simeq g$ if $f \preceq g$ and $f \succeq g$ (i.e., $f \circ g(x) = g \circ f(x)$ for any $x \in \mathbb{R}$), and $f \prec g$ (or $g \succ f$) if $f \preceq g$ and $f \not\succeq g$.

Note that the relation \preceq is not *total relation* in general, here a relation \preceq is called total if $f \preceq g$ or $g \preceq f$ for any f, g . For example, let $f_1(x) = \max\{2x, 3x\}$ and $f_2(x) = \max\{2x - 1, 3x + 1\}$. Then $f_1 \circ f_2(0) (= 3)$ is greater than $f_2 \circ f_1(0) (= 1)$, but $f_1 \circ f_2(-2) (= -10)$ is less than $f_2 \circ f_1(-2) (= -9)$.

However, if two consecutive functions are comparable, then we have the following easy but useful lemma.

► **Lemma 6.** *Let f_1, \dots, f_n be monotone nondecreasing functions. If $f_i \preceq f_{i+1}$, then it holds that $f_n \circ \dots \circ f_{i+2} \circ f_{i+1} \circ f_i \circ f_{i-1} \circ \dots \circ f_1(x) \geq f_n \circ \dots \circ f_{i+2} \circ f_i \circ f_{i+1} \circ f_{i-1} \circ \dots \circ f_1(x)$ for any $x \in \mathbb{R}$.*

It follows from the lemma that, for monotone functions f_i , there exists a maximum total composition ordering $f_n \circ f_{n-1} \circ \dots \circ f_1$ that satisfies $f_1 \preceq f_2 \preceq \dots \preceq f_n$, if the relation is total. Moreover, if the relation \preceq is in addition transitive (i.e., $f \preceq g$ and $g \preceq h$ imply $f \preceq h$), then it is not difficult to see that $f_1 \preceq f_2 \preceq \dots \preceq f_n$ becomes a sufficient condition that $f_n \circ f_{n-1} \circ \dots \circ f_1$ is a maximum total composition ordering, where the proof is given as the more general form in Lemma 8.

The relation is total if all functions are linear or of the form $\max\{ax + b, x\}$ with $a \geq 0$.

► **Lemma 7.** *The relation \preceq is total for linear functions.*

Proof. Let $f_i(x) = a_i x + b_i$ and $f_j(x) = a_j x + b_j$. Then we have

$$\begin{aligned} f_i \preceq f_j &\iff f_i \circ f_j(x) \leq f_j \circ f_i(x) \text{ for any } x \in \mathbb{R} \\ &\iff a_i(a_j x + b_j) + b_i \leq a_j(a_i x + b_i) + b_j \text{ for any } x \in \mathbb{R} \\ &\iff b_i(1 - a_j) \leq b_j(1 - a_i). \end{aligned} \tag{1}$$

Since the last inequality consists of only constants, we have $f_i \preceq f_j$ or $f_i \succeq f_j$. ◀

When all functions are of the form $\max\{ax + b, x\}$ with $a \geq 0$, the totality of the relation is proven in Lemma 11.

We further note that the relation \preceq is transitive for linear functions $f(x) = ax + b$ with $a > 1$, since (1) is equivalent to $b_i/(1 - a_i) \leq b_j/(1 - a_j)$, and hence the ordering $b_1/(1 - a_1) \leq b_2/(1 - a_2) \leq \dots \leq b_n/(1 - a_n)$ gives an optimal solution for the maximum total composition ordering problem. Therefore, it can be solved efficiently by sorting the elements by $b_i/(1 - a_i)$. The same statement holds when all linear functions have slope less than 1. This idea is used for the linear deterioration and linear shortening models for time-dependent scheduling problems. However, in general, this is not the case, i.e., the relation \preceq does not satisfy transitivity. Let $f_1(x) = 2x + 1$, $f_2(x) = 2x - 1$, and $f_3(x) = x/2$. Then we have $f_1 \prec f_2$, $f_2 \prec f_3$, and $f_3 \prec f_1$, which implies that the transitivity is not satisfied for linear functions, and $\bar{f}_1 \prec \bar{f}_2$, $\bar{f}_2 \prec \bar{f}_3$, and $\bar{f}_3 \prec \bar{f}_1$ hold, implying that the transitivity is not satisfied for the functions of the form $\max\{ax + b, x\}$ with $a \geq 0$. These show that the maximum total and partial composition ordering problems are not trivial, even when all functions are monotone and linear.

We first show the following key lemma which can be used even for *non-transitive* relations.

► **Lemma 8.** *For monotone nondecreasing functions $f_i : \mathbb{R} \rightarrow \mathbb{R}$ ($i \in [n]$), if a permutation $\sigma : [n] \rightarrow [n]$ satisfies that $i \leq j$ implies $f_{\sigma(i)} \preceq f_{\sigma(j)}$ for any $i, j \in [n]$, then σ is an optimal solution for the maximum total composition ordering problem $((f_i)_{i \in [n]}, c)$.*

Proof. Without loss of generality, we may assume that σ is the identity permutation. Let σ' be an optimal solution for the maximum total composition ordering problem such that it has the minimum inversion number, where the inversion number denotes the number of pairs (i, j) with $i < j$ and $\sigma'(i) > \sigma'(j)$. Then we show that σ' is the identity permutation by contradiction. Assume that $\sigma'(l) > \sigma'(l + 1)$ for some l . Then consider the following permutation:

$$\tau(i) = \begin{cases} \sigma'(i) & (i \neq l, l + 1), \\ \sigma'(l + 1) & (i = l), \\ \sigma'(l) & (i = l + 1). \end{cases}$$

Since $\sigma'(l + 1) < \sigma'(l)$ implies $f_{\sigma'(l+1)} \preceq f_{\sigma'(l)}$ by the condition of the identity σ , Lemma 6 implies that τ is also optimal for the problem. Since τ has an inversion number smaller than the one for σ' , we derive a contradiction. Therefore, σ' is the identity. ◀

As mentioned above, if the relation \preceq is in addition transitive (i.e., \preceq is a total preorder), then such a σ always exists.

To efficiently solve the maximum partial composition ordering problem for the linear functions, we show that for $\bar{f}_i(x) = \max\{a_i x + b_i, x\}$ ($a_i \geq 0$), (i) there exists a permutation σ which satisfies the condition in Lemma 8 and (ii) the permutation σ can be computed efficiently. Let us analyze the relation \preceq in terms of the following γ .

► **Definition 9.** For a linear function $f(x) = ax + b$, we define

$$\gamma(f) = \begin{cases} \frac{b}{1-a} & (a \neq 1), \\ +\infty & (a = 1 \text{ and } b < 0), \\ -\infty & (a = 1 \text{ and } b \geq 0). \end{cases}$$

Note that $\gamma(f)$ is the solution of the equation $f(x) = x$ if $\gamma(f) \neq -\infty, +\infty$.

In the rest of the paper, we assume without loss of generality that no f_i is the identity (i.e., $f_i(x) = x$), since we can ignore identity function for both the total and partial composition problems.

► **Lemma 10.** *Let $f_i(x) = a_i x + b_i$ and $f_j(x) = a_j x + b_j$ be (non-identity) monotone nondecreasing functions. Then we have the following statements;*

- (a) *if $a_i, a_j = 1$, then $f_i \simeq f_j$,*
- (b) *if $a_i, a_j \geq 1$ and $a_i \cdot a_j > 1$, then $f_i \preceq f_j \Leftrightarrow \gamma(f_i) \leq \gamma(f_j)$,*
- (c) *if $a_i, a_j < 1$, then $f_i \preceq f_j \Leftrightarrow \gamma(f_i) \leq \gamma(f_j)$,*
- (d) *if $a_i \geq 1, a_j < 1$, then $f_i \preceq f_j \Leftrightarrow \gamma(f_i) \geq \gamma(f_j)$ and $f_i \succeq f_j \Leftrightarrow \gamma(f_i) \leq \gamma(f_j)$.*

► **Lemma 11.** *For (non-identity) monotone nondecreasing linear functions $f_i(x) = a_i x + b_i$ and $f_j(x) = a_j x + b_j$, we have the following statements;*

- (a) *if $a_i, a_j \geq 1$ and $\gamma(f_i) \leq \gamma(f_j)$, then $\bar{f}_i \preceq \bar{f}_j$,*
- (b) *if $a_i, a_j < 1$ and $\gamma(f_i) \leq \gamma(f_j)$, then $\bar{f}_i \preceq \bar{f}_j$,*
- (c) *if $a_i < 1, a_j \geq 1$, and $\gamma(f_i) \leq \gamma(f_j)$, then $\bar{f}_i \simeq \bar{f}_j$,*
- (d) *if $a_i \geq 1, a_j < 1$, and $\gamma(f_i) \leq \gamma(f_j)$, then $\bar{f}_i \succeq \bar{f}_j$.*

Note that Lemma 11 implies that the relation \preceq is total for the functions of the form $\max\{ax + b, x\}$ with $a \geq 0$. Moreover, it implies that the following permutation σ satisfies the condition in Lemma 8.

For a linear function $f(x) = ax + b$, let $\delta(f) = +1$ if $a \geq 1$ and otherwise $\delta(f) = -1$. Let $\sigma : [n] \rightarrow [n]$ denote a permutation that is *compatible* with the lexicographic ordering with respect to $(\delta(f_i), \gamma(f_i))$, i.e., $(\delta(f_{\sigma(i)}), \gamma(f_{\sigma(i)}))$ is lexicographically smaller than or equal to $(\delta(f_{\sigma(j)}), \gamma(f_{\sigma(j)}))$ if $i < j$. Namely, there exists an integer k such that $0 \leq k \leq n$, $\delta(f_{\sigma(1)}) = \dots = \delta(f_{\sigma(k)}) = -1$, $\delta(f_{\sigma(k+1)}) = \dots = \delta(f_{\sigma(n)}) = +1$, $\gamma(f_{\sigma(1)}) \leq \dots \leq \gamma(f_{\sigma(k)})$, and $\gamma(f_{\sigma(k+1)}) \leq \dots \leq \gamma(f_{\sigma(n)})$. Then we have the following lemma by Lemma 11.

► **Lemma 12.** *For (non-identity) monotone nondecreasing linear functions f_i ($i \in [n]$), let σ denote a permutation compatible with the lexicographic order with respect to $(\delta(f_i), \gamma(f_i))$. Then $i \leq j$ implies $\bar{f}_{\sigma(i)} \preceq \bar{f}_{\sigma(j)}$ for any $i, j \in [n]$.*

By Lemmas 8 and 12, the maximum partial composition ordering problem for the monotone nondecreasing linear functions f_i , equivalently, the maximum total composition ordering problem for the functions \bar{f}_i such that f_i 's are monotone nondecreasing linear functions can be solved by computing the lexicographic order with respect to $(\delta(f_i), \gamma(f_i))$. Therefore, it can be solved in $O(n \log n)$ time, which proves the partial composition part of Theorem 1. We remark that the time complexity $O(n \log n)$ of the problem is the best possible in the comparison model. We also remark that the optimal value for the maximum partial composition ordering problem for $f_i(x) = a_i x + b_i$ ($a_i \geq 0$) forms a piecewise linear function (in c) with at most $(n + 1)$ pieces.

We next extend this tractability result to Theorem 4. For $i \in [n]$, let $h_i(x) = a_i x + b_i$ be a monotone nondecreasing linear function, and let $f_i(x) = \max\{h_i(x), c_i\}$ for a constant c_i . We consider the maximum partial composition ordering problem for f_i 's. As mentioned in the introduction, the problem includes the two-valued free-order secretary problem, and it is a generalization of the maximum partial composition ordering problem for monotone linear functions.

We instead consider the maximum total composition ordering problem for the functions

$$\bar{f}_i(x) = \max\{a_i x + b_i, c_i, x\} \text{ for } a_i \in \mathbb{R}_+, b_i, c_i \in \mathbb{R}. \tag{2}$$

► **Lemma 13.** *Let $c \in \mathbb{R}$, and let \bar{f}_i ($i \in [n]$) be a function defined as (2). Then there exists an optimal solution σ for the maximum total composition ordering problem $((\bar{f}_i)_{i \in [n]}, c)$ such that no i (> 1) satisfies $\bar{h}_{\sigma(i)} \circ \bar{f}_{\sigma(i-1)} \circ \cdots \circ \bar{f}_{\sigma(1)}(c) < c_{\sigma(i)}$, where $h_i(x) = a_i x + b_i$.*

Proof. Let σ denote an optimal solution for the problem. Assume that there exists an index i that satisfies the condition in the lemma. Let i^* denote the largest such i . Then by the definition of i^* , we have $\bar{f}_{\sigma(i^*)} \circ \cdots \circ \bar{f}_{\sigma(1)}(c) = \bar{f}_{\sigma(i^*)}(c) = c_{\sigma(i^*)}$. It holds that $c_{\sigma(i)} < c_{\sigma(i^*)}$ for any i with $0 \leq i < i^*$, since $c_{\sigma(i)} \leq \bar{f}_{\sigma(i)} \circ \cdots \circ \bar{f}_{\sigma(1)}(c) \leq \bar{f}_{\sigma(i^*-1)} \circ \cdots \circ \bar{f}_{\sigma(1)}(c) < c_{\sigma(i^*)}$, where $c_{\sigma(0)} = c$ is assumed. Thus, we have $\bar{f}_{\sigma(n)} \circ \cdots \circ \bar{f}_{\sigma(1)}(c) = \bar{f}_{\sigma(n)} \circ \cdots \circ \bar{f}_{\sigma(i^*)}(c) \leq \bar{f}_{\sigma(i^*-1)} \circ \cdots \circ \bar{f}_{\sigma(1)} \circ \bar{f}_{\sigma(n)} \circ \cdots \circ \bar{f}_{\sigma(i^*)}(c)$. This implies that $(\sigma(i^*), \dots, \sigma(n), \sigma(1), \dots, \sigma(i^*-1))$ is also an optimal permutation for the problem. Moreover, in the composition according to this permutation, the constant part of f_i ($i \neq i^*$) is not explicitly used by the definition of i^* and $c_{\sigma(i)} < c_{\sigma(i^*)}$ for any i ($< i^*$), which completes the proof. ◀

Proof of Theorem 4. It follows from Lemma 13 that an optimal solution for the problem can be obtained by solving the following $n + 1$ instances of the maximum partial composition ordering problem for monotone nondecreasing linear functions $((h_i)_{i \in [n]}, c)$ and $((h_i)_{i \in [n] \setminus \{k\}}, c_k)$ for all $k \in [n]$.

Therefore, we have an $O(n^2 \log n)$ -time algorithm by directly applying Theorem 1 to the problem. Moreover, we note that the maximum partial composition ordering problem for monotone nondecreasing linear functions can be solved in linear time, if we know the lexicographic ordering with respect to (δ, γ) . This implies that the problem can be solved in $O(n^2)$ time by first computing the lexicographic order with respect to $(\delta(h_i), \gamma(h_i))$. ◀

3 Maximum Total Composition Ordering Problem

In this section we prove the total composition part of Theorem 1 and Theorem 2.

The following lemma shows the relationships between $\gamma(f_i)$, $\gamma(f_j)$, $\gamma(f_j \circ f_i)$ and $\gamma(f_i \circ f_j)$ for monotone linear functions.

► **Lemma 14.** *For monotone nondecreasing linear functions $f_i(x) = a_i x + b_i$ and $f_j(x) = a_j x + b_j$ ($a_i, a_j \geq 0$), we have the following statements.*

- (a) *If $\gamma(f_i) = \gamma(f_j)$, then $\gamma(f_i) = \gamma(f_j) = \gamma(f_j \circ f_i)$,*
- (b) *If $\gamma(f_i) < \gamma(f_j)$ and $a_i, a_j \geq 1$, then $\gamma(f_i) \leq \gamma(f_j \circ f_i) \leq \gamma(f_j)$,*
- (c) *If $\gamma(f_i) < \gamma(f_j)$ and $a_i, a_j < 1$, then $\gamma(f_i) \leq \gamma(f_j \circ f_i) \leq \gamma(f_j)$,*
- (d) *If $\gamma(f_i) < \gamma(f_j)$, $a_i < 1$, $a_j \geq 1$, and $a_i \cdot a_j \geq 1$, then $\gamma(f_j \circ f_i) \geq \gamma(f_j)$ ($> \gamma(f_i)$),*
- (e) *If $\gamma(f_i) < \gamma(f_j)$, $a_i < 1$, $a_j \geq 1$, and $a_i \cdot a_j < 1$, then $\gamma(f_j \circ f_i) \leq \gamma(f_i)$ ($< \gamma(f_j)$),*
- (f) *If $\gamma(f_i) < \gamma(f_j)$, $a_i \geq 1$, $a_j < 1$, and $a_i \cdot a_j \geq 1$, then $\gamma(f_j \circ f_i) \leq \gamma(f_i)$ ($< \gamma(f_j)$),*
- (g) *If $\gamma(f_i) < \gamma(f_j)$, $a_i \geq 1$, $a_j < 1$, and $a_i \cdot a_j < 1$, then $\gamma(f_j \circ f_i) \geq \gamma(f_j)$ ($> \gamma(f_i)$).*

By Lemmas 10 and 14, we have the following inequalities for compositions of four functions.

► **Lemma 15.** *For monotone nondecreasing linear functions $f_i(x) = a_i x + b_i$ ($i = 1, 2, 3, 4$), if $a_1, a_3 \geq 1$, $a_2, a_4 < 1$ and $\gamma(f_1) \geq \gamma(f_2) \geq \gamma(f_3) \geq \gamma(f_4)$, then we have*

$$f_4 \circ f_3 \circ f_2 \circ f_1(x) \leq \max\{f_4 \circ f_1 \circ f_3 \circ f_2(x), f_3 \circ f_2 \circ f_4 \circ f_1(x)\} \text{ for any } x.$$

► **Lemma 16.** *For monotone nondecreasing linear functions $f_i(x) = a_i x + b_i$ ($i = 1, 2, 3, 4$), if $a_1, a_3 < 1$, $a_2, a_4 \geq 1$ and $\gamma(f_1) \geq \gamma(f_2) \geq \gamma(f_3) \geq \gamma(f_4)$, then we have*

$$f_4 \circ f_3 \circ f_2 \circ f_1(x) \leq \max\{f_4 \circ f_1 \circ f_3 \circ f_2(x), f_3 \circ f_2 \circ f_4 \circ f_1(x)\} \text{ for any } x.$$

Proof. We only prove Lemma 15 since the proof of Lemma 16 is similar. Let $g(x) = f_3 \circ f_2(x)$. If $a_2 \cdot a_3 \geq 1$, then $\gamma(g) \leq \gamma(f_3) \leq \gamma(f_1)$ holds by 1 and 6 in Lemma 14, and $g \circ f_1(x) \leq f_1 \circ g(x)$ holds by 1 and 2 in Lemma 10. Thus, we have $f_4 \circ f_3 \circ f_2 \circ f_1(x) \leq f_4 \circ f_1 \circ f_3 \circ f_2(x)$.

On the other hand, if $a_2 \cdot a_3 < 1$, then $\gamma(g) \geq \gamma(f_2) \geq \gamma(f_4)$ holds by 1 and 7 in Lemma 14, and $f_4 \circ g(x) \leq g \circ f_4(x)$ holds by 3 in Lemma 10. Thus, we have $f_4 \circ f_3 \circ f_2 \circ f_1(x) \leq f_3 \circ f_2 \circ f_4 \circ f_1(x)$. ◀

By Lemmas 15 and 16, we obtain the following lemma.

► **Lemma 17.** *There exists an optimal permutation σ for the maximum total composition ordering problem for monotone nondecreasing functions f_i ($i \in [n]$) such that at most two i 's satisfy $\delta(f_{\sigma(i)}) \cdot \delta(f_{\sigma(i+1)}) = -1$.*

Next, we provide inequalities for compositions of three functions.

► **Lemma 18.** *For monotone nondecreasing linear functions $f_i(x) = a_i x + b_i$ ($i = 1, 2, 3$), if $a_1, a_3 \geq 1$, $a_2 < 1$, $a_1 \cdot a_2 \cdot a_3 \geq 1$ and $\gamma(f_1) \geq \gamma(f_2) \geq \gamma(f_3)$, then we have*

$$f_3 \circ f_2 \circ f_1(x) \leq \max\{f_2 \circ f_1 \circ f_3(x), f_1 \circ f_3 \circ f_2(x)\} \text{ for any } x.$$

► **Lemma 19.** *For monotone nondecreasing linear functions $f_i(x) = a_i x + b_i$ ($i = 1, 2, 3$), if $a_1, a_3 < 1$, $a_2 \geq 1$, $a_1 \cdot a_2 \cdot a_3 < 1$ and $\gamma(f_1) \geq \gamma(f_2) \geq \gamma(f_3)$, then we have*

$$f_3 \circ f_2 \circ f_1(x) \leq \max\{f_2 \circ f_1 \circ f_3(x), f_1 \circ f_3 \circ f_2(x)\} \text{ for any } x.$$

Proof. We only prove Lemma 18 since the proof of Lemma 19 is similar. If $a_2 \cdot a_3 \geq 1$, then $\gamma(f_3 \circ f_2) \leq \gamma(f_3) \leq \gamma(f_1)$ by 1 and 6 in Lemma 14, and it implies $f_3 \circ f_2 \circ f_1(x) \leq f_1 \circ f_3 \circ f_2(x)$ by 1 and 2 in Lemma 10. If $a_2 \cdot a_3 < 1$ and $\gamma(f_3 \circ f_2) \geq \gamma(f_1)$, then $f_3 \circ f_2 \circ f_1(x) \leq f_1 \circ f_3 \circ f_2(x)$ by 4 in Lemma 10.

If $a_1 \cdot a_2 \geq 1$, then $\gamma(f_2 \circ f_1) \geq \gamma(f_1) \geq \gamma(f_3)$ by 1 and 4 in Lemma 14, and it implies $f_3 \circ f_2 \circ f_1(x) \leq f_2 \circ f_1 \circ f_3(x)$ by 1 and 2 in Lemma 10. If $a_1 \cdot a_2 < 1$ and $\gamma(f_2 \circ f_1) \leq \gamma(f_3)$, then $f_3 \circ f_2 \circ f_1(x) \leq f_2 \circ f_1 \circ f_3(x)$ by 4 in Lemma 10.

Otherwise, we have $a_2 \cdot a_3 < 1$, $a_1 \cdot a_2 < 1$, $\gamma(f_3 \circ f_2) < \gamma(f_1)$, and $\gamma(f_2 \circ f_1) > \gamma(f_3)$. Then we have $\gamma((f_3 \circ f_2) \circ f_1) \geq \gamma(f_1)$ by 4 in Lemma 14, and $\gamma(f_3 \circ (f_2 \circ f_1)) \leq \gamma(f_3)$ by 6 in Lemma 14 since $a_1 \cdot a_2 \cdot a_3 \geq 1$. Therefore $\gamma(f_1) = \gamma(f_2) = \gamma(f_3)$. This together with $\gamma(f_3 \circ f_2) < \gamma(f_1)$ contradicts 1 in Lemma 14. ◀

By Lemmas 10, 14, 17, 18, and 19, we get the following lemmas.

► **Lemma 20.** *If $\prod_{i=1}^n a_i \geq 1$, then there exists an optimal permutation σ such that, for some two integers s, t ($0 \leq s \leq t \leq n$), $\delta(f_{\sigma(t+1)}) = \dots = \delta(f_{\sigma(n)}) = \delta(f_{\sigma(1)}) = \dots = \delta(f_{\sigma(s)}) = -1$, $\delta(f_{\sigma(s+1)}) = \dots = \delta(f_{\sigma(t)}) = 1$, $\gamma_{\sigma(t+1)} \leq \dots \leq \gamma_{\sigma(n)} \leq \gamma_{\sigma(1)} \leq \dots \leq \gamma_{\sigma(s)}$, and $\gamma_{\sigma(s+1)} \leq \dots \leq \gamma_{\sigma(t)}$.*

► **Lemma 21.** *If $\prod_{i=1}^n a_i < 1$, then there exists an optimal permutation σ such that, for some two integers s, t ($0 \leq s \leq t \leq n$), $\delta(f_{\sigma(t+1)}) = \dots = \delta(f_{\sigma(n)}) = \delta(f_{\sigma(1)}) = \dots = \delta(f_{\sigma(s)}) = 1$, $\delta(f_{\sigma(s+1)}) = \dots = \delta(f_{\sigma(t)}) = -1$, $\gamma_{\sigma(t+1)} \leq \dots \leq \gamma_{\sigma(n)} \leq \gamma_{\sigma(1)} \leq \dots \leq \gamma_{\sigma(s)}$, and $\gamma_{\sigma(s+1)} \leq \dots \leq \gamma_{\sigma(t)}$.*

Proof. We only prove Lemma 20 since the proof of Lemma 21 is similar. By Lemma 17, there exists an optimal permutation σ and two integers s, t ($0 \leq s \leq t \leq n$) such that

42:10 Optimal Composition Ordering Problems for Piecewise Linear Functions

$\delta(f_{\sigma(1)}) = \cdots = \delta(f_{\sigma(s)}) = -\delta(f_{\sigma(s+1)}) = \cdots = -\delta(f_{\sigma(t)}) = \delta(f_{\sigma(t+1)}) = \cdots = \delta(f_{\sigma(n)})$. By Lemma 10, we have

$$\gamma_{\sigma(1)} \leq \cdots \leq \gamma_{\sigma(s)}, \gamma_{\sigma(s+1)} \leq \cdots \leq \gamma_{\sigma(t)}, \gamma_{\sigma(t+1)} \leq \cdots \leq \gamma_{\sigma(n)}.$$

This implies that the lemma holds when $s = 0$ or $t = n$. For $0 < s \leq t < n$, we separately consider the following two cases.

Case 1: If $\delta(f_{\sigma(s+1)}) = \cdots = \delta(f_{\sigma(t)}) = +1$, let $g = f_{\sigma(n-1)} \circ \cdots \circ f_{\sigma(2)}$. Then Lemma 10 and the optimality of σ imply $\gamma(f_{\sigma(1)}) \geq \gamma(g) \geq \gamma(f_{\sigma(n)})$, since $-\delta(f_{\sigma(1)}) = \delta(g) = -\delta(f_{\sigma(n)}) = +1$. This proves the lemma.

Case 2: If $\delta(f_{\sigma(s+1)}) = \cdots = \delta(f_{\sigma(t)}) = -1$, then let $h_1 = f_{\sigma(s)} \circ \cdots \circ f_{\sigma(1)}$, $h_2 = f_{\sigma(t)} \circ \cdots \circ f_{\sigma(s+1)}$ and $h_3 = f_{\sigma(n)} \circ \cdots \circ f_{\sigma(t+1)}$. If $\gamma(h_1) < \gamma(h_2)$, then $h_3 \circ h_2 \circ h_1(x) \leq h_3 \circ h_1 \circ h_2(x)$ by 4 in Lemma 10. If $\gamma(h_2) < \gamma(h_3)$, then $h_3 \circ h_2 \circ h_1(x) \leq h_2 \circ h_3 \circ h_1(x)$ by 4 in Lemma 10. Otherwise (i.e., $\gamma(h_1) \geq \gamma(h_2) \geq \gamma(h_3)$), we have

$$h_3 \circ h_2 \circ h_1(x) \leq \max\{h_2 \circ h_1 \circ h_3(x), h_1 \circ h_3 \circ h_2(x)\}$$

by Lemma 18. In either case, we can obtain a desired optimal solution by modifying σ . ◀

By Lemmas 20 and 21, we obtain polynomial time algorithm for the maximum total composition ordering problem for monotone nondecreasing linear functions.

Proof of the total composition part of Theorem 1. By Lemmas 20 and 21, the total composition ordering problem for monotone nondecreasing linear functions can be computed as follows. Let $\sigma : [n] \rightarrow [n]$ be a permutation which satisfies $\delta(f_{\sigma(1)}) = \cdots = \delta(f_{\sigma(r)}) = -1$, $\delta(\sigma(r+1)) = \cdots = \delta(f_{\sigma(n)})$, $\gamma(f_{\sigma(1)}) \leq \cdots \leq \gamma(f_{\sigma(r)})$, and $\gamma(f_{\sigma(r+1)}) \leq \cdots \leq \gamma(f_{\sigma(n)})$. Then Lemmas 20 and 21 implies that there exists an optimal solution of the form $(\sigma(t), \sigma(t+1), \dots, \sigma(n), \sigma(1), \sigma(2), \dots, \sigma(t-1))$ for some t . Therefore, the problem can be computed in polynomial time by checking n permutations above. To reduce the time complexity, let $d_k = f_{\sigma(k-1)} \circ \cdots \circ f_{\sigma(1)} \circ f_{\sigma(n)} \circ \cdots \circ f_{\sigma(k)}(c)$ for $k = 1, \dots, n$. Let $a = \prod_{i=1}^n a_i$. Then it is not difficult to see that $d_{k+1} = a_{\sigma(k)} \cdot (d_k - a \cdot c) - b_{\sigma(k)} \cdot (a - 1) + a \cdot c$, and hence the problem is solvable in $O(n \log n)$ time. ◀

Next, we prove Theorem 2. We use dynamic programming to find the optimal value.

Proof of Theorem 2. Without loss of generality, we may assume that the indices of functions are $\delta(f_1) = \cdots = \delta(f_r) = -1$, $\delta(f_{r+1}) = \cdots = \delta(f_n) = 1$, $\gamma(f_1) \leq \cdots \leq \gamma(f_r)$, and $\gamma(f_{r+1}) \leq \cdots \leq \gamma(f_n)$. We use dynamic programming to solve the problem. Let $m(i, j, l)$ be the maximum value of $f_{\sigma(l)} \circ \cdots \circ f_{\sigma(1)}(c)$ for a permutation σ such that $i \leq \sigma(1) < \sigma(2) < \cdots < \sigma(l) \leq i + j - 1$ if $i + j - 1 \leq n$, and $i \leq \sigma(1) < \cdots < \sigma(p) \leq n$, $1 \leq \sigma(p+1) < \cdots < \sigma(l) \leq i + j - 1 - n$ for some p ($0 \leq p \leq l$) if $i + j - 1 > n$. We claim that the optimal value for the problem is $\max_{i=1}^n m(i, n, k)$.

Let $\sigma^* : [n] \rightarrow [n]$ be an optimal permutation for the problem. By Lemmas 20 and 21, we can assume that $i^* \leq \sigma^*(1) < \cdots < \sigma^*(p) \leq n$, $1 \leq \sigma^*(p+1) < \cdots < \sigma^*(k) \leq i^* - 1$ for some i^* and p . Therefore, we have $f_{\sigma^*(k)} \circ \cdots \circ f_{\sigma^*(1)}(c) \leq m(i^*, n, k) \leq \max_{i=1}^n m(i, n, k)$ and thus $\max_{i=1}^n m(i, n, k)$ is the optimal value for the problem.

For each i, j, l , the value $m(i, j, l)$ satisfies the following relation:

$$m(i, j, l) = \begin{cases} c & (l = 0), \\ f_{\sigma(j)}(m(i, j - 1, l - 1)) & (l \geq 1, j = l), \\ \max\{m(i, j - 1, l), f_{\sigma(j)}(m(i, j - 1, l - 1))\} & (l \geq 1, j > l). \end{cases}$$

To evaluate $\max_{i=1}^n m(i, n, k)$, our algorithm calculate the values of $m(i, j, l)$ for $0 \leq i, j \leq n$ and $0 \leq l \leq k$. Therefore, we can obtain the optimal value for the problem in $O(k \cdot n^2)$ time. ◀

4 Negative Results for the Optimal Composition Ordering Problems

In the previous sections, we show that both the total and partial composition ordering problems can be solved efficiently if all f_i 's are monotone linear. It turns out that they cannot be generalized to nonlinear functions f_i . In this section, we show the optimal composition ordering problems are in general intractable, even if all f_i 's are monotone increasing, piecewise linear functions with at most two pieces. We remark that the maximum total composition ordering problem is known to be NP-hard, even if all f_i 's are monotone increasing, *concave*, piecewise linear functions with at most two pieces [4], which can be shown by considering the time-dependent scheduling problem.

Due to space limitation, we only provide the result for the concave case.

The concave case

In this section, we consider the case in which all f_i 's are monotone increasing, concave, piecewise linear functions with at most two pieces, that is, f_i is given as $f_i(x) = \min\{a_i^1 x + b_i^1, a_i^2 x + b_i^2\}$ for some reals a_i^1, a_i^2, b_i^1 and b_i^2 with $a_i^1, a_i^2 > 0$. For our reductions, we use the PARTITION problem, which is known to be NP-complete [10].

PARTITION: Given n positive integers a_1, \dots, a_n with $\sum_{i=1}^n a_i = 2T$, ask whether exists a subset $I \subseteq [n]$ such that $\sum_{i \in I} a_i = T$.

Proof for Theorem 3(i). We show that PARTITION can be reduced to the problem. Let a_1, \dots, a_n denote positive integers with $\sum_{i=1}^n a_i = 2T$. We construct $n + 2$ functions f_i ($i = 1, \dots, n + 2$) as follows:

$$f_i(x) = \begin{cases} x + a_i & \text{if } i = 1, \dots, n, \\ \min\{2x, \frac{1}{2}x + \frac{3}{2}T\} & \text{if } i = n + 1, \\ 6\alpha T(x - (3T - \frac{1}{2})) + (3T - \frac{1}{2}) & \text{if } i = n + 2. \end{cases}$$

It is clear that all f_i 's are monotone, concave, and piecewise linear with at most two pieces. Moreover, we note that all f_i 's ($i = 1, \dots, n + 1$) satisfy $f_i(x) \geq x$ if $0 \leq x \leq 3T$, and $f_{n+2}(x) \leq x$ if $x \leq 3T - 1/2$. We claim that $3T$ is the optimal value for the maximum partial (total) composition ordering problem for f_i ($i = 1, \dots, n + 1$) and $c = 0$ if there exists a partition $I \subseteq [n]$ such that $\sum_{i \in I} a_i = T$, and the optimal value is at most $3T - 1/2$ if $\sum_{i \in I} a_i \neq T$ for any partition $I \subseteq [n]$. This implies that the optimal value for the maximum partial (total) composition ordering problem for f_i ($i = 1, \dots, n + 2$) and $c = 0$ is at least $3\alpha T$ if $\sum_{i \in I} a_i = T$ for some $I \subseteq [n]$, and at most $3T$ if $\sum_{i \in I} a_i \neq T$ for any partition $I \subseteq [n]$, since $f_{n+2}(3T) = 3\alpha T + 3T - 1/2 > 3\alpha T$ and $f_{n+2}(x) \leq x$ if $x \leq 3T - 1/2$. Thus, there exists no α -approximation algorithm for the problems unless P=NP.

Let $\sigma : [n + 1] \rightarrow [n + 1]$ denote a permutation with $\sigma(l) = n + 1$. Then define $I = \{\sigma(i) : i = 1, \dots, l - 1\}$ and $q = \sum_{i \in I} a_i$. Note that $\sum_{i=l+1}^{n+1} a_{\sigma(i)} = \sum_{i \notin I} a_i = 2T - q$. Consider the function composition by σ :

$$\begin{aligned} & f_{\sigma(n+1)} \circ \dots \circ f_{\sigma(l+1)} \circ f_{\sigma(l)} \circ f_{\sigma(l-1)} \circ \dots \circ f_{\sigma(1)}(0) \\ &= f_{\sigma(n)} \circ \dots \circ f_{\sigma(l+1)} \circ f_{n+1}(q) \\ &= f_{\sigma(n)} \circ \dots \circ f_{\sigma(l+1)} \left(\min \left\{ 2q, \frac{1}{2}q + \frac{3}{2}T \right\} \right) \\ &= \min \left\{ 2q, \frac{1}{2}q + \frac{3}{2}T \right\} + 2T - q = \min \left\{ q, -\frac{1}{2}q + \frac{3}{2}T \right\} + 2T. \end{aligned}$$

Note that $\min \left\{ q, -\frac{1}{2}q + \frac{3}{2}T \right\} \leq T$ holds, where the equality holds only when $q = T$. This implies that

$$f_{\sigma(n+1)} \circ \dots \circ f_{\sigma(l+1)} \circ f_{\sigma(l)} \circ f_{\sigma(l-1)} \circ \dots \circ f_{\sigma(1)}(0) \begin{cases} = 3T & (q = T), \\ \leq 3T - 1/2 & (q \neq T) \end{cases} \quad (3)$$

since q is an integer. This proves the claim. \blacktriangleleft

References

- 1 M. Babaioff, N. Immorlica, D. Kempe, and R. Kleinberg. A knapsack secretary problem with applications. In *Proceedings of APPROX/RANDOM 2007*, pages 16–28, 2007.
- 2 M. Babaioff, N. Immorlica, and R. Kleinberg. Matroids, secretary problems, and online mechanisms. In *Proceedings of SODA 2007*, pages 434–443, 2007.
- 3 J.-Y. Cai, P. Cai, and Y. Zhu. On a scheduling problem of time deteriorating jobs. *J. Complexity*, 14(2):190–209, 1998.
- 4 T. C. E. Cheng and Q. Ding. The complexity of scheduling starting time dependent tasks with release times. *Inf. Process. Lett.*, 65(2):75–79, 1998.
- 5 T. C. E. Cheng, Q. Ding, M. Y. Kovalyov, A. Bachman, and A. Janiak. Scheduling jobs with piecewise linear decreasing processing times. *Naval Res. Logist.*, 50(6):531–554, 2003.
- 6 T. C. E. Cheng, Q. Ding, and B. M. T. Lin. A concise survey of scheduling with time-dependent processing times. *EJOR*, 152(1):1–13, 2004.
- 7 B. Dean, M. Goemans, and J. Vondrák. Adaptivity and approximation for stochastic packing problems. In *Proceedings of SODA 2005*, pages 395–404, 2005.
- 8 B. Dean, M. Goemans, and J. Vondrák. Approximating the stochastic knapsack problem: the benefit of adaptivity. *Math. of OR*, 33(4):945–964, 2008.
- 9 T. S. Ferguson. Who solved the secretary problem? *Statist. Sci.*, 4(3):282–289, 1989.
- 10 M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman New York, 1979.
- 11 S. Gawiejnowicz. Scheduling deteriorating jobs subject to job or machine availability constraints. *EJOR*, 180(1):472–478, 2007.
- 12 S. Gawiejnowicz. *Time-Dependent Scheduling*. Springer, 2008.
- 13 S. Gawiejnowicz and L. Pankowska. Scheduling jobs with varying processing times. *Inf. Process. Lett.*, 54(3):175–178, 1995.
- 14 J. N. Gupta and S. K. Gupta. Single facility scheduling with nonlinear processing times. *Computers & Industrial Engineering*, 14(4):387–393, 1988.
- 15 K. I.-J. Ho, J. Y.-T. Leung, and W.-D. Wei. Complexity of scheduling tasks with time-dependent execution times. *Inf. Process. Lett.*, 48(6):315–320, 1993.
- 16 Y. Kawase, K. Makino, and K. Seimi. Optimal composition ordering problems for piecewise linear functions. *CoRR*, abs/1601.05480, 2016.

- 17 O. I. Melnikov and Y. M. Shafransky. Parametric problem of scheduling theory. *Cybernetics*, 15:352–357, 1980.
- 18 S. Oveis Gharan and J. Vondrák. On variants of the matroid secretary problem. In *Proceedings of ESA 2011*, pages 335–346, 2011.
- 19 V. S. Tanaev, V. S. Gordon, and Y. M. Shafransky. *Scheduling Theory: Single-Stage Systems*. Kluwer Academic Publishers, 1994.
- 20 W. Wajs. Polynomial algorithm for dynamic sequencing problem. *Archiwum Automatyki i Telemekhaniki*, 31(3):209–213, 1986.