J. Math. Anal. Appl. 348 (2008) 346–355

Contents lists available at ScienceDirect

Journal of Mathematical Analysis and Applications

www.elsevier.com/locate/jmaa

Applications of the complexity space to the General Probabilistic Divide and Conquer Algorithms

L.M. García-Raffi [a,1], S. Romaguera [a,*,2], M.P. Schellekens [b,3]

[a] Instituto Universitario de Matemática Pura y Aplicada, Universidad Politécnica de Valencia, 46071 Valencia, Spain

[b] Centre of Efficiency-Oriented Languages, Department of Computer Science, National University of Ireland, Western Road, Cork, Ireland

ARTICLE INFO

Article history:
Received 16 April 2008
Available online 16 July 2008
Submitted by B. Cascales

Keywords:
Complexity quasi-metric space
Smyth complete
Fixed point
Improver
Recurrence
Probabilistic Divide and Conquer Algorithm

ABSTRACT

Schellekens [M. Schellekens, The Smyth completion: A common foundation for denotational semantics and complexity analysis, in: Proc. MFPS 11, in: Electron. Notes Theor. Comput. Sci., vol. 1, 1995, pp. 535–556], and Romaguera and Schellekens [S. Romaguera, M. Schellekens, Quasi-metric properties of complexity spaces, Topology Appl. 98 (1999) 311–322] introduced a topological foundation to obtain complexity results through the application of Semantic techniques to Divide and Conquer Algorithms. This involved the fact that the complexity (quasi-metric) space is Smyth complete and the use of a version of the Banach fixed point theorem and improver functionals. To further bridge the gap between Semantics and Complexity, we show here that these techniques of analysis, based on the theory of complexity spaces, extend to General Probabilistic Divide and Conquer schema discussed by Flajolet [P. Flajolet, Analytic analysis of algorithms, in: W. Kuich (Ed.), 19th Internat. Colloq. ICALP'92, Vienna, July 1992; Automata, Languages and Programming, in: Lecture Notes in Comput. Sci., vol. 623, 1992, pp. 186–210]. In particular, we obtain a general method which is useful to show that for several recurrence equations based on the recursive structure of General Probabilistic Divide and Conquer Algorithms, the associated functionals have a unique fixed point which is the solution for the corresponding recurrence equation.

1. Introduction

In [13], M. Schellekens introduced the theory of complexity (quasi-metric) spaces in order to provide a topological foundation for the complexity analysis of algorithms. Application of these spaces to the complexity analysis of Divide and Conquer Algorithms were given in Section 4 of [13]. This forms part of the programme to bridge Complexity and Semantics, a research challenge posed at the IFIP2000 International Conference on Exploring New Frontiers of Theoretical Informatics. This topic has since be the focus of various conferences. S. Romaguera and M. Schellekens obtained in [10] several quasi-metric properties of the complexity spaces which are interesting from a computational point of view, via the analysis of the so-called dual complexity (quasi-metric) space. In particular, the (dual) complexity space is Smyth complete. Moreover, the dual complexity space can be modelled as a (quasi-)normed semilinear space as it is shown in [11]. Further contributions to the study of the mathematical structure of this space may be found in [4,9,12].

* Corresponding author.
E-mail addresses: lmgarcia@mat.upv.es (L.M. García-Raffi), sromague@mat.upv.es (S. Romaguera), m.schellekens@cs.ucc.ie (M.P. Schellekens).

[1] Supported by the Spanish Ministry of Education and Science and FEDER, under grant MTM2006-11690-C02-01.
[2] Supported by the Spanish Ministry of Education and Science and FEDER, under grant MTM2006-14925-C02-01.
[3] Supported by the Science Foundation Ireland Investigator Award 07/IN.1/I977.

0022-247X/$ – see front matter
doi:10.1016/j.jmaa.2008.07.026

Throughout this paper the letters $\mathbb{N}$ and ω will denote the set of natural numbers and nonnegative integer numbers, respectively. Our basic references for quasi-uniform and quasi-metric spaces are [2] and [6].

Following the modern terminology, by a quasi-metric on a set X we mean a nonnegative real valued function d on $X \times X$ such that for all $x, y, z \in X$: (i) $d(x, y) = d(y, x) = 0 \Leftrightarrow x = y$; and (ii) $d(x, y) \leqslant d(x, z) + d(z, y)$.

A quasi-metric space is a pair (X, d) such that X is a (nonempty) set and d is a quasi-metric on X.

If d is a quasi-metric on X, then the function d^s defined on $X \times X$ by $d^s(x, y) = d(x, y) \vee d(y, x)$, for all $x, y \in X$, is a metric on X.

M.B. Smyth presented in [15] and [16] a topological framework for denotational semantics based on theory of complete (and totally bounded) quasi-uniform and quasi-metric spaces. This study was continued, among other authors, by Ph. Sünderhauf [18] and H.P.A. Künzi [5]. The following notion and characterization will be enough for our purposes here.

A sequence $(x_n)_n$ in a quasi-metric space (X, d) is said to be left K-Cauchy [7,8] if for each $\varepsilon > 0$ there is $n_0 \in \mathbb{N}$ such that $d(x_n, x_m) < \varepsilon$ whenever $m \geqslant n \geqslant n_0$. A quasi-metric space (X, d) is Smyth complete if and only if every left K-Cauchy sequence is convergent with respect to the topology induced by the metric d^s [10].

As usual, by a recurrence equation (a recurrence relation, or simply a recurrence) we mean an equation that defines a sequence recursively: each term of the sequence is defined as a function of the preceding terms. If $T(n)$, $n \in \mathbb{N}$, is the nth term of the sequence, then the recurrence equation will be simply denoted by T. By a solution of T we mean a function f defined on $\mathbb{N}$ and that satisfies the recurrence equation for all $n \in \mathbb{N}$.

A typical example of a recurrence equation is the equation which specifies the factorial function f given by $f(0) = 1$, and $f(n) = nf(n-1)$ for all $n \in \mathbb{N}$.

This recurrence equation of course determines the factorial sequence $(n!)_n$. As such the solution of the equation is a function $f : \omega \to \mathbb{N}$.

The theory of Complexity Spaces, that we will recall next, relies on an adaptation of techniques of Denotational Semantics to the context of Complexity Theory. Denotational Semantics is an approach which allows the formalization of the meanings of programming languages by constructing mathematical objects (called denotations) which describe the meanings of expressions from the languages. This enables one to specify the way in which programs, written in the programming language under consideration, compute in an unambiguous way. In the absence of Denotational Semantics, i.e., the case where programming languages are specified in natural language, ambiguities arise and different programmers may implement the language in different ways. Hence there is no guarantee on portability of software. Denotational semantics originated in the work of D. Scott and C. Strachey in the 1960's. As originally developed by Strachey and Scott, denotational semantics provided the denotation (meaning) of a computer program as a function that mapped inputs to outputs (see [14,17]). To give denotations to recursively defined programs, Scott proposed working with continuous functions between domains, specifically complete partial orders. A main tool in Denotational Semantics is to associate a functional with recursive programs. If we reconsider the recurrence equation discussed above (which can be viewed as code for a recursive program computing the factorial function) then we can associate the functional Φ defined by $\Phi f(0) = 1$, and $\Phi f(n) = nf(n-1)$ for all $n \in \mathbb{N}$. Then, the functional Φ transforms the function f into a new function Φf defined by the scheme displayed above.

The formalization of the meaning of the recursive program, in our case the program computing the factorial function, is then obtained as the least fixed point of this functional Φ over the function space under consideration.

Similarly, for each recurrence equation, a functional Φ can be associated with this recurrence.

Observe that we consider functionals over functions from ω to $\mathbb{N}$. This is a deliberate simplification. In Denotational Semantics, the function spaces are so-called domains (e.g., complete partial orders) and the functions typically are partial, i.e., they have function domains and ranges which can include the symbol $\perp$ to indicate the undefined case.

The theory of Complexity Spaces, discussed next, has been introduced to capture the complexity of the programs. For instance the number of steps the program takes during its computation could be captured, as opposed to a representation of the input-output relation of the program as is typically the case in Denotational Semantics. In the theory of Complexity Spaces, complexity functions are formally represented as unique fixed points of functionals associated with the recurrence equations that specify the complexity of a program.

Let us recall [13], that the complexity (quasi-metric) space is the pair $(\mathcal{C}, d_{\mathcal{C}})$, where

$$\mathcal{C} = \left\{ f : \omega \to (0, \infty]:\ \sum_{n=0}^{\infty} 2^{-n} \frac{1}{f(n)} < \infty \right\},$$

and $d_{\mathcal{C}}$ is the quasi-metric on $\mathcal{C}$ given by

$$d_{\mathcal{C}}(f, g) = \sum_{n=0}^{\infty} 2^{-n} \left(\left(\frac{1}{g(n)} - \frac{1}{f(n)} \right) \vee 0 \right),$$

for all $f, g \in \mathcal{C}$. (We adopt the convention that $1/\infty = 0$.)

The elements of $\mathcal{C}$ are called complexity functions.

It is well known [10] that the quasi-metric space $(\mathcal{C}, d_{\mathcal{C}})$ is Smyth complete.

In the sequel we denote by $\mathcal{C}_0$ the set $\{f \in \mathcal{C}:\ f(n) < \infty \text{ for all } n \in \omega\}$.

The applicability of the theory of complexity spaces to the complexity analysis of Divide and Conquer Algorithms was illustrated in Section 7 of [13], by a new proof, based on the Banach fixed point theorem and improver functionals, of the fact that mergesort has optimal asymptotic average running time.

Following Definition 6.2 of [13], a functional Φ from $(\mathcal{C}, d_{\mathcal{C}})$ into itself, is an improver with respect to a function $f \in \mathcal{C}_0$ if for each $n \in \omega$, $\Phi^{n+1} f \leqslant \Phi^n f$.

Note that if Φ is monotone increasing (i.e., $\Phi f \leqslant \Phi g$ whenever $f \leqslant g$), to show that Φ is an improver with respect to f, it suffices to verify that $\Phi f \leqslant f$.

The intuition is that an improver is a functional that corresponds to a transformation on algorithms and satisfies the following condition: the iterative applications of the transformation to a given algorithm yield an improved algorithm at each step of the iteration.

In this paper we discuss the complexity analysis of Probabilistic Divide and Conquer Algorithm. We obtain a general theorem that permits us to show that for many recurrence equations based on the recursive structure of Probabilistic Divide and Conquer Algorithm, the associated functionals have a unique fixed point which is the solution for our recurrence equation. Our technique is based on constructing a monotone increasing functional Φ, associated with a given recurrence equation T, for which there is a complexity function g such that $g \leqslant \Phi g$, and, by Smyth completeness of $(\mathcal{C}, d_{\mathcal{C}})$, the sequence $(\Phi^k g)_k$ of iterations is convergent in $(\mathcal{C}, (d_{\mathcal{C}})^s)$ to some $f_T \in \mathcal{C}$ which is the unique fixed point of Φ and, then, it is the solution for the recurrence equation T. Moreover, if Φ is an improver with respect to some $g \in \mathcal{C}_0$, then $f_T \leqslant g$ and thus f_T is in class of g, i.e., $f_T(n) \in \mathcal{O}(g(n))$.

Since the Probabilistic Divide and Conquer Algorithm provides an example of an algorithm which does not have a recurrence equation of the Divide and Conquer kind, the techniques of analysis based on the theory of complexity spaces are applicable to more general classes of algorithms than the class of Divide and Conquer Algorithms.

2. Fixed points for functionals on the complexity space associated with recurrence equations

In this section we prove a general theorem on existence of fixed points for functionals, defined on the complexity space, which appear in a natural way in the analysis of Probabilistic Divide and Conquer Algorithm.

We first state some auxiliary results which will be crucial in obtaining our main result (Theorem 2).

Theorem 1. *(See [10].) The complexity space is Smyth complete.*

Proposition 1. *Let $\Phi : \mathcal{C} \to C$ be monotone increasing. If there is $g \in \mathcal{C}$ such that $g \leqslant \Phi g$, then*:

(1) *There is $f \in \mathcal{C}$ such that $\lim_{k\to\infty} (d_{\mathcal{C}})^s(f, \Phi^k g) = 0$.*
(2) *$\Phi^k g \leqslant f \leqslant \Phi f$ for all $k \in \omega$.*

Proof. (1) Since $g \leqslant \Phi g$ and Φ is monotone increasing, it follows that

$$\Phi^k g \leqslant \Phi^{k+1} g,$$

for all $k \in \omega$. Therefore

$$d_{\mathcal{C}}\big(\Phi^k g, \Phi^{k+1} g\big) = 0,$$

for all $k \in \omega$. Hence $(\Phi^k g)_k$ is a left K-Cauchy sequence in $(\mathcal{C}, d_{\mathcal{C}})$. By Theorem 1 there is $f \in \mathcal{C}$ such that

$$\lim_{k\to\infty} (d_{\mathcal{C}})^s\big(f, \Phi^k g\big) = 0.$$

(2) Fix $k \in \omega$. Given $\varepsilon > 0$, by (1), there is $j \geqslant k$ such that $(d_{\mathcal{C}})^s(f, \Phi^j g) < \varepsilon$. Thus

$$d_{\mathcal{C}}\big(\Phi^k g, f\big) \leqslant d_{\mathcal{C}}\big(\Phi^k g, \Phi^j g\big) + d_{\mathcal{C}}\big(\Phi^j g, f\big) = d_{\mathcal{C}}\big(\Phi^j g, f\big) < \varepsilon.$$

Consequently $d_{\mathcal{C}}(\Phi^k g, f) = 0$, so that, $\Phi^k g \leqslant f$.

Finally, since Φ is monotone increasing, we have $\Phi^{k+1} g \leqslant \Phi f$ for all $k \in \omega$, and hence

$$d_{\mathcal{C}}(f, \Phi f) \leqslant d_{\mathcal{C}}\big(f, \Phi^{k+1} g\big) + d_{\mathcal{C}}\big(\Phi^{k+1} g, \Phi f\big) = d_{\mathcal{C}}\big(f, \Phi^{k+1} g\big),$$

for all $k \in \omega$. It follows that $d_{\mathcal{C}}(f, \Phi f) = 0$, i.e., $f \leqslant \Phi f$. The proof is finished. □

Proposition 2. *Let $(f_k)_k$ be a sequence in $\mathcal{C}_0$. If there is $f \in \mathcal{C}_0$ such that $\lim_{k\to\infty} (d_{\mathcal{C}})^s(f, f_k) = 0$, then $(f_k)_k$ is pointwise convergent to f with respect to the Euclidean metric, i.e., for each $n \in \omega$ and each $\varepsilon > 0$ there is $k_0 \in \mathbb{N}$ such that $|f(n) - f_k(n)| < \varepsilon$ for all $k \geqslant k_0$.*

Proof. Fix $n \in \omega$. Choose $\varepsilon > 0$. By hypothesis, there is $k_0 \in \mathbb{N}$ such that

$$\left|\frac{1}{f(n)} - \frac{1}{f_k(n)}\right| < \frac{\varepsilon}{f(n)(f(n)+\varepsilon)},$$

for all $k \geqslant k_0$. Hence

$$\left|f(n) - f_k(n)\right| < \frac{\varepsilon f_k(n)}{f(n)+\varepsilon}, \tag{$\star$}$$

and

$$\frac{1}{f(n)} - \frac{\varepsilon}{f(n)(f(n)+\varepsilon)} < \frac{1}{f_k(n)},$$

for all $k \geqslant k_0$. From the last inequality it follows that $f_k(n) < f(n) + \varepsilon$, for all $k \geqslant k_0$. We immediately deduce from ($\star$) that

$$\left|f(n) - f_k(n)\right| < \varepsilon,$$

for all $k \geqslant k_0$. □

Theorem 2. *Let T be a recurrence equation. Suppose that there is $n_0 \geqslant 2$ such that*

$$T(n) = u(n) + \sum_{k=1}^{n-1} v_k(n)T(k),$$

for all $n \geqslant n_0$, where $u \in \mathcal{C}_0$ and $(v_k)_k$ is a sequence of positive functions on $\mathbb{N}$ satisfying the following condition.

There is $K > 0$ such that for all $n > n_0$,

$$\sum_{k=n_0}^{n-1} v_k(n) \leqslant K. \tag{$*$}$$

Then, the functional $\Phi : \mathcal{C} \to C$ defined, for all $f \in \mathcal{C}$, by

$$\Phi f(0) = T(1), \qquad \Phi f(n) = T(n), \quad n = 1, \ldots, n_0 - 1,$$

and

$$\Phi f(n) = u(n) + \sum_{k=1}^{n-1} v_k(n) f(k),$$

for all $n \geqslant n_0$, has a unique fixed point $f_T \in \mathcal{C}_0$ which is obviously the solution of the recurrence equation T.

Furthermore, if Φ is an improver with respect to some $g \in \mathcal{C}$, it follows that $f_T \leqslant g$.

Proof. First we observe that, indeed, $\Phi f \in \mathcal{C}$ for all $f \in \mathcal{C}$, because $u \in \mathcal{C}$ and thus

$$\sum_{n=0}^{\infty} 2^{-n} \frac{1}{\Phi f(n)} \leqslant \sum_{n=0}^{\infty} 2^{-n} \frac{1}{u(n)} < \infty.$$

Moreover Φ is monotone increasing since for $f, g \in \mathcal{C}$ with $f \leqslant g$, one has

$$\Phi f(n) = \Phi g(n),$$

for $n = 0, 1, \ldots, n_0 - 1$, and

$$\Phi f(n) \leqslant u(n) + \sum_{k=1}^{n-1} v_k(n) g(k) = \Phi g(n),$$

for all $n \geqslant n_0$.

Consider the function $g : \omega \to (0, \infty)$ defined by $g(0) = T(1)$, $g(n) = T(n)$, $n = 1, \ldots, n_0 - 1$, and $g(n) = u(n)$ for all $n \geqslant n_0$. Since $u \in \mathcal{C}_0$, it follows that $g \in \mathcal{C}_0$. Furthermore $g(n) = \Phi g(n)$ for $n = 0, 1, \ldots, n_0 - 1$, and clearly $g(n) \leqslant \Phi g(n)$ for all $n \geqslant n_0$, by the construction of Φ. Therefore, we can apply Proposition 1, so that:

(1) There is $f_T \in \mathcal{C}$ such that $\lim_{k\to\infty} (d_{\mathcal{C}})^s (f_T, \Phi^k g) = 0$, and
(2) $\Phi^k g \leqslant f_T \leqslant \Phi f_T$ for all $k \in \omega$.

Note that by the construction of Φ and the fact that $g \in \mathcal{C}_0$, we easily deduce that $\Phi^k g \in \mathcal{C}_0$ for all $k \in \omega$.

Now we show that $f_T \in \mathcal{C}_0$. Indeed, assume the contrary, and let j be the first nonnegative integer number such that $f_T(j) = \infty$. Since $f_T \leqslant \Phi f_T$, it follows that $\Phi f_T(j) = \infty$ and $j \geqslant n_0$. Therefore

$$\Phi f_T(j) = u(j) + \sum_{k=1}^{j-1} v_k(j) f_T(k) < \infty,$$

a contradiction. Hence $f_T \in \mathcal{C}_0$.

Then, we can apply Proposition 2, and thus, the sequence $(\Phi^k g)_k$ is pointwise convergent to f_T.

Next we prove that $f_T = \Phi f_T$. To this end, we first recall that, for $n = 0, 1, \ldots, n_0 - 1$, we have $\Phi f_T(n) = \Phi g(n)$. So by condition (2) and the definition of g, we deduce that $f_T(n) = \Phi f_T(n) = g(n)$, for $n = 0, 1, \ldots, n_0 - 1$. Hence $\Phi g(n_0) = \Phi f_T(n_0)$. It follows from (2) that $f_T(n_0) = \Phi f_T(n_0)$.

Now fix $n > n_0$. Choose an arbitrary $\varepsilon > 0$. Then, there is $j \in \mathbb{N}$ such that

$$\left| f_T(k) - \Phi^j g(k) \right| < \varepsilon, \quad k = n_0, \ldots, n-1.$$

Consequently, we obtain (recall that, by condition (2), $f_T(n) = \Phi^j g(n)$, $n = 0, 1, \ldots, n_0 - 1$):

$$\begin{aligned}\Phi f_T(n) &= u(n) + \sum_{k=1}^{n-1} v_k(n) f_T(k) < u(n) + \sum_{k=1}^{n_0-1} v_k(n) f_T(k) + \sum_{k=n_0}^{n-1} v_k(n)\big(\varepsilon + \Phi^j g(k)\big) \\ &= \varepsilon \sum_{k=n_0}^{n-1} v_k(n) + \Phi^{j+1} g(n) \leqslant K\varepsilon + f_T(n).\end{aligned}$$

Therefore $\Phi f_T(n) \leqslant f_T(n)$. We conclude that $\Phi f_T = f_T$, so f_T is a fixed point of Φ, and thus it is a solution for the recurrence equation T.

In order to show that f_T is the unique fixed point of Φ, we suppose that there is $f'_T \in \mathcal{C}$ with $\Phi f'_T = f'_T$. Then $f'_T(n) = f_T(n)$ for $n = 0, 1, \ldots, n_0 - 1$, so by the construction of Φ, $\Phi f'_T(n_0) = \Phi f_T(n_0)$, i.e., $f'_T(n_0) = f_T(n_0)$, and inductively, $f'_T(n) = f_T(n)$ for all $n > n_0$.

Finally, suppose that Φ is an improver with respect to some $g \in \mathcal{C}$. Then $\Phi g \leqslant g$ (see Section 1), so, $f_T(n) = \Phi g(n) \leqslant g(n)$ for $n = 0, 1, \ldots, n_0 - 1$. Hence, we obtain

$$f_T(n_0) = u(n_0) + \sum_{k=1}^{n_0-1} v_k(n_0) f_T(k) \leqslant u(n_0) + \sum_{k=1}^{n_0-1} v_k(n_0) g(k) = \Phi g(n_0) \leqslant g(n_0),$$

and, by induction on n, we easily deduce that $f_T(n) \leqslant g(n)$ for $n > n_0$. This concludes the proof. □

Remark 1. If the recurrence T verifies $T(1) = 0$ and $T(n) > 0$ for all $n \geqslant 2$, we may construct the recurrence S given by $S(n) = T(n+1)$ for all $n \in \mathbb{N}$. Then, by Theorem 2, the recurrence S has a (unique) solution $f_S \in \mathcal{C}_0$. Thus, the function $f_T : \mathbb{N} \to [0, \infty)$ given by $f_T(1) = 0$ and $f_T(n) = f_S(n-1)$ for all $n \geqslant 2$ is the (unique) solution of T. This observation will be used in Section 3.

3. The complexity analysis of Probabilistic Divide and Conquer Algorithm: Examples

When discussing the analysis of Probabilistic Divide and Conquer Algorithm by means of recurrences, the following general recurrence equation is obtained:

$$T(n) = p(n) + \sum_{k=1}^{n-1} q(n,k) T(k),$$

where $T(1) \geqslant 0$, $p(n) = c_1 n + c_2$, with $c_1 > 0$ and $2c_1 + c_2 > 0$, and the $q(n,k)$'s are nonnegative and proportional to the splitting probabilities that express the changes that a task of size n involve a subtask of size $k < n$. (Observe that $T(2) > 0$ because $2c_1 + c_2 > 0$, and, thus, $T(n) > 0$ for all $n \geqslant 2$.)

The following are typical examples for $q(n,k)$.

$$\text{(A)}\quad \frac{\alpha}{n}, \qquad \text{(B)}\quad \frac{2\alpha(n-k)}{n(n+1)}, \qquad \text{(C)}\quad \frac{\alpha}{n}\sum_{j=k+1}^{n}\frac{1}{j}, \qquad \text{(D)}\quad \frac{2\alpha(k-1)(n-k)}{n(n-1)(n-2)},$$

with $\alpha > 0$, which arise in Probabilistic Divide and Conquer Algorithm or binary search trees, fully specified search in $2-d$ quadtress, partial match queries in $2-d$ quadtress, and median-of-three Quicksort, respectively (see [1, Section 4], and [3, p. 609]).

Next we discuss the complexity analysis of Probabilistic Divide and Conquer Algorithm for the four cases cited above.

Case (A). The recurrence equation T is given by

$$T(n) = c_1 n + c_2 + \frac{\alpha}{n}\sum_{k=1}^{n-1} T(k),$$

for all $n \geqslant 2$.

We may assume $T(1) > 0$ (see Remark 1), and thus T satisfies the conditions of Theorem 2, with $n_0 = 2$, $u \in \mathcal{C}_0$ with $u(0) = u(1) = c > 0$, c arbitrary, and $u(n) = c_1 n + c_2$ for all $n \geqslant 2$, and $v_k(n) = \alpha/n$ for all $k \in \mathbb{N}$ (observe that $\sum_{k=2}^{n-1} v_k(n) = \alpha(n-2)/n < \alpha$ for all $n > 2$).

Consequently, the recurrence equation T has a unique solution $f_T \in \mathcal{C}_0$.

Now we will obtain a class of complexity functions for which the functional Φ associated with T, is an improver. To this end, we write for each $n \geqslant 2$:

$$T(n+1) = c_1(n+1) + c_2 + \frac{\alpha}{n+1}\sum_{k=1}^{n} T(k) = c_1(n+1) + c_2 + \frac{\alpha}{n+1}\left(T(n) + \sum_{k=1}^{n-1} T(k)\right)$$
$$= c_1(n+1) + c_2 + \frac{\alpha}{n+1}\left(T(n) + \frac{n}{\alpha}\big(T(n) - (c_1 n + c_2)\big)\right) = h(n+1) + \frac{n+\alpha}{n+1}T(n),$$

where

$$h(n+1) = c_1(n+1) + c_2 - \frac{n(c_1 n + c_2)}{n+1} = \frac{c_1(2n+1) + c_2}{n+1},$$

for all $n \geqslant 2$.

Therefore, we have

$$T(2) = 2c_1 + c_2 + \frac{\alpha}{2}T(1),$$

and

$$T(n) = h(n) + \frac{n+\alpha-1}{n}T(n-1),$$

for all $n \geqslant 3$.

Hence Φ can be expressed by $\Phi f(0) = \Phi f(1) = T(1)$, $\Phi f(2) = T(2)$, and

$$\Phi f(n) = h(n) + \frac{n+\alpha-1}{n} f(n-1),$$

for all $n \geqslant 3$, with

$$h(n) = \frac{c_1(2n-1) + c_2}{n}.$$

Then, for $g \in \mathcal{C}$ satisfying $T(n) \leqslant g(n)$, $n = 0, 1, 2$, and

$$\frac{c_1(2n-1) + c_2}{n} + \frac{n+\alpha-1}{n} g(n-1) \leqslant g(n), \quad n \geqslant 3, \tag{I}$$

it follows that $\Phi g \leqslant g$, so Φ is an improver with respect to g, and hence, the solution f_T of the recurrence equation T verifies $f_T \leqslant g$ by Theorem 2.

Next we apply these methods to deduce the known fact that, for $0 < \alpha \leqslant 2$, $f_T(n) \in \mathcal{O}(n \log_a n)$, for any $a > 1$.

Indeed, we first observe that given $K, r > 0$ and $a > 1$, a double application of the L'Hôpital rule gives

$$\lim_{x \to +\infty} \frac{K[x^2(\log_a x - \log_a(x-1)) + \log_a(x-1)]}{rx + s} = \frac{K}{r \cdot \ln a}.$$

Therefore, if $K > r \cdot \ln a$, there exists $n_0 \in \mathbb{N}$ such that for each $n \geqslant n_0$,

$$Kn \log_a n > \frac{(n+1)}{n} K(n-1)\log_a(n-1) + \frac{rn + s}{n}. \tag{$**$}$$

By applying the inequality $(**)$ to the particular case that $r = 2c_1$ and $s = c_2 - c_1$, we obtain for $0 < \alpha \leqslant 2$, that

$$Kn \log_a n > \frac{(n+\alpha-1)}{n} K(n-1)\log_a(n-1) + \frac{c_1(2n-1) + c_2}{n},$$

whenever $n \geqslant n_0$. So, inequality (I) above is satisfied for $g(n) = Kn\log_a n$, with $K > 2c_1 \cdot \ln a$, and $n \geqslant n_0$.

We conclude that $f_T(n) \in \mathcal{O}(n \log_a n)$, whenever $0 < \alpha \leqslant 2$.

Case (B). The recurrence equation T is given by

$$T(n) = c_1 n + c_2 + \frac{2\alpha}{n(n+1)}\sum_{k=1}^{n-1}(n-k)T(k),$$

for all $n \geqslant 2$.

As in Case (A) we may assume that $T(1) > 0$, and thus T satisfies the conditions of Theorem 2 with $n_0 = 2$, $u \in \mathcal{C}_0$ with $u(0) = u(1) = c > 0$, c arbitrary, and $u(n) = c_1 n + c_2$ for all $n \geqslant 2$, and $v_k(n) = 2\alpha(n-k)/n(n+1)$ for all $k \in \mathbb{N}$ (observe that $\sum_{k=2}^{n-1}(n-k) = (n^2 - 3n + 2)/2$, so $\sum_{k=2}^{n-1} v_k(n) = \alpha(n^2 - 3n + 2)/n(n+1) < \alpha(n-2)/n + 1$ for all $n > 2$).

Consequently, the recurrence equation T has a unique solution $f_T \in \mathcal{C}_0$.

Now we will obtain a class of complexity functions for which the functional Φ associated with T is an improver. To this end, we write for each $n \geqslant 2$:

$$\begin{aligned}
T(n+1) &= c_1(n+1) + c_2 + \frac{2\alpha}{(n+1)(n+2)} \sum_{k=1}^{n} (n+1-k)T(k) \\
&= c_1(n+1) + c_2 + \frac{2\alpha}{(n+1)(n+2)} \left(\sum_{k=1}^{n-1} (n-k)T(k) + \sum_{k=1}^{n} T(k) \right) \\
&= c_1(n+1) + c_2 + \frac{2\alpha}{(n+1)(n+2)} \left(\frac{n(n+1)}{2\alpha} \big(T(n) - (c_1 n + c_2) \big) + \sum_{k=1}^{n} T(k) \right) \\
&= \left(c_1(n+1) + c_2 - \frac{n(c_1 n + c_2)}{n+2} \right) + \frac{n}{n+2} T(n) + \frac{2\alpha}{(n+1)(n+2)} \sum_{k=1}^{n} T(k).
\end{aligned}$$

Therefore

$$T(2) = 2c_1 + c_2 + \frac{\alpha}{3} T(1)$$

and

$$T(n) = h(n) + \frac{n-1}{n+1} T(n-1) + \frac{2\alpha}{n(n+1)} \sum_{k=1}^{n-1} T(k),$$

for all $n \geqslant 3$, where

$$h(n) = c_1 n + c_2 - \frac{(n-1)(c_1(n-1) + c_2)}{n+1} = \frac{c_1(3n-1) + 2c_2}{n+1}.$$

Hence Φ can be expressed by

$$\Phi f(0) = \Phi f(1) = T(1), \qquad \Phi f(2) = T(2), \qquad \Phi f(3) = h(3) + \frac{1}{2} T(2) + \frac{\alpha}{6} \big(T(1) + T(2) \big),$$

and

$$\Phi f(n) = h(n) + \frac{n-1}{n+1} f(n-1) + \frac{2\alpha}{n(n+1)} \left(T(1) + T(2) + \sum_{k=3}^{n-1} f(k) \right),$$

for all $n \geqslant 4$.

Then, for $g \in \mathcal{C}$ monotone increasing (i.e., $g(n) \leqslant g(n+1)$ for all $n \in \omega$) satisfying $T(n) \leqslant g(n)$, $n = 0, 1, 2, 3$, and

$$\frac{c_1(3n-1) + 2c_2}{n+1} + \frac{(n-1)(n+2\alpha)}{n(n+1)} g(n-1) \leqslant g(n), \quad n \geqslant 4, \tag{II}$$

it follows that $\Phi g \leqslant g$, so Φ is an improver with respect to g, and hence, the solution f_T of the recurrence equation T verifies $f_T \leqslant g$ by Theorem 2.

Similarly to Case (A), we show that for $0 < \alpha \leqslant 3/2$, $f_T(n) \in \mathcal{O}(n \log_a n)$, for any $a > 1$.

Indeed, note that for each $n \in \mathbb{N}$ we have $n + 1 > (n-1)(n+3)/(n+1)$. Hence, it immediately follows from inequality $(**)$ that there is $n_0 \in \mathbb{N}$ such that for each $n \geqslant n_0$, and $0 < \alpha \leqslant 3/2$,

$$Kn \log_a n > \frac{(n-1)(n+2\alpha)}{n(n+1)} K(n-1) \log_a(n-1) + \frac{c_1(3n-1) + 2c_2}{n+1},$$

where $K > 3c_1 \cdot \ln a$.

Consequently, inequality (II) above is satisfied for $g(n) = Kn \log_a n$, with $K > 3c_1 \cdot \ln a$, and $n \geqslant n_0$.

We conclude that $f_T(n) \in \mathcal{O}(n \log_a n)$, whenever $0 < \alpha \leqslant 3/2$.

Case (C). The recurrence equation T is given by

$$T(n) = c_1 n + c_2 + \frac{\alpha}{n} \sum_{k=1}^{n-1} (H_n - H_k) T(k),$$

for all $n \geqslant 2$, where $H_n = \sum_{j=1}^{n} 1/j$.

In this case, the boundedness condition (∗) of Theorem 2 can be easily obtained by developing the terms of our recurrence as follows. For each $n \geqslant 2$, put:

$$\begin{aligned} T(n+1) &= c_1(n+1) + c_2 + \frac{\alpha}{n+1}\sum_{k=1}^{n}(H_{n+1} - H_k)T(k) \\ &= c_1(n+1) + c_2 + \frac{\alpha}{n+1}\left(\sum_{k=1}^{n-1}(H_n - H_k)T(k) + \frac{1}{n+1}\sum_{k=1}^{n}T(k)\right) \\ &= c_1(n+1) + c_2 + \frac{\alpha}{n+1}\left(\frac{n}{\alpha}\big(T(n) - (c_1 n + c_2)\big) + \frac{1}{n+1}\sum_{k=1}^{n}T(k)\right) \\ &= \left(c_1(n+1) + c_2 - \frac{n(c_1 n + c_2)}{n+1}\right) + \frac{n}{n+1}T(n) + \frac{\alpha}{(n+1)^2}\sum_{k=1}^{n}T(k). \end{aligned}$$

Therefore

$$T(n) = h(n) + \left(\frac{n-1}{n} + \frac{\alpha}{n^2}\right)T(n-1) + \frac{\alpha}{n^2}\sum_{k=1}^{n-2}T(k),$$

for all $n \geqslant 3$, where

$$h(n) = c_1 n + c_2 - \frac{(n-1)(c_1(n-1) + c_2)}{n} = \frac{c_1(2n-1) + c_2}{n},$$

i.e., $h(n) = u(n)$ for $n \geqslant 3$.

Now, as in Case (A) we may assume that $T(1) > 0$, and thus T satisfies the conditions of Theorem 2. Indeed, we have $n_0 = 3$, $u(n) = c_1(2n-1) + c_2/n$ for all $n \geqslant 3$, and

$$v_k(n) = \frac{\alpha}{n^2}, \quad k < n-1, \qquad v_{n-1}(n) = \frac{\alpha}{n^2} + \frac{n-1}{n}.$$

Then $u \in \mathcal{C}_0$ and for $n \geqslant 4$,

$$\sum_{k=3}^{n-1} v_k(n) \leqslant \frac{\alpha(n-3)}{n^2} + \frac{\alpha}{n^2} + \frac{n-1}{n} < 2\alpha + 1.$$

Consequently, by Theorem 2, the recurrence equation T has a unique solution $f_T \in \mathcal{C}_0$.

Next we will obtain a class of complexity functions for which the functional Φ associated with T is an improver. In fact, we have for $f \in \mathcal{C}$

$$\Phi f(n) = h(n) + \left(\frac{n-1}{n} + \frac{\alpha}{n^2}\right)f(n-1) + \frac{\alpha}{n^2}\sum_{k=1}^{n-2}f(k), \quad n \geqslant 3.$$

Then, for $g \in \mathcal{C}$ monotone increasing satisfying $T(n) \leqslant g(n)$, $n = 0, 1, 2$, and

$$\frac{c_1(2n-1) + c_2}{n} + \frac{(n-1)(n+\alpha)}{n^2}g(n-1) \leqslant g(n), \quad n \geqslant 3, \tag{III}$$

it follows that $\Phi g \leqslant g$, so Φ is an improver with respect to g, and hence, the solution f_T of the recurrence equation T verifies $f_T \leqslant g$ by Theorem 2.

Similarly to Cases (A) and (B), we show that for $0 < \alpha \leqslant 2$, $f_T(n) \in \mathcal{O}(n \log_a n)$, for any $a > 1$.

Indeed, note that for each $n \in \mathbb{N}$ we have $n + 1 > (n-1)(n+2)/n$. Hence, it immediately follows from inequality (∗∗) that there is $n_0 \in \mathbb{N}$ such that for each $n \geqslant n_0$, and $0 < \alpha \leqslant 2$,

$$Kn\log_a n > \frac{(n-1)(n+\alpha)}{n^2}K(n-1)\log_a(n-1) + \frac{c_1(2n-1) + c_2}{n},$$

where $K > 2c_1 \cdot \ln a$.

So, inequality (III) above is satisfied for $g(n) = Kn\log_a n$, with $K > 2c_1 \cdot \ln a$, and $n \geqslant n_0$.

We conclude that $f_T(n) \in \mathcal{O}(n\log_a n)$, whenever $0 < \alpha \leqslant 2$.

Case (D). The recurrence equation T is given by

$$T(n) = c_1 n + c_2 + \frac{2\alpha}{n(n-1)(n-2)}\sum_{k=1}^{n-1}(k-1)(n-k)T(k),$$

for all $n \geqslant 3$.

As in Case (A) we may assume that $T(1) > 0$ and thus T satisfies the conditions of Theorem 2, with $n_0 = 3$, $u \in \mathcal{C}_0$ with $u(0) = u(1) = u(2) = c > 0$, c arbitrary, and $u(n) = c_1 n + c_2$ for all $n \geqslant 3$, and $v_k(n) = 2\alpha(k-1)(n-k)/n(n-1)(n-2)$ for all $k \in \mathbb{N}$ (observe that $\sum_{k=3}^{n-1}(k-1)(n-k) = (n-3)(n^2-4)/6$, so $\sum_{k=3}^{n-1} v_k(n) < \alpha/3$ for all $n > 3$).

Consequently, the recurrence equation T has a unique solution $f_T \in \mathcal{C}_0$.

Now we will obtain a class of complexity functions for which the functional Φ associated with T is an improver. To this end, we write for each $n \geqslant 2$:

$$T(n+1) = c_1(n+1) + c_2 + \frac{2\alpha}{(n+1)n(n-1)} \sum_{k=1}^{n}(k-1)(n+1-k)T(k)$$
$$= h(n+1) + \frac{n-2}{n+1}T(n) + \frac{2\alpha}{(n-1)n(n+1)} \sum_{k=2}^{n}(k-1)T(k),$$

where

$$h(n+1) = c_1(n+1) + c_2 - \frac{(n-2)(c_1 n + c_2)}{n+1}.$$

Therefore

$$T(n) = h(n) + \frac{n-3}{n}T(n-1) + \frac{2\alpha}{(n-2)(n-1)n} \sum_{k=2}^{n-1}(k-1)T(k),$$

for all $n \geqslant 3$.

Hence Φ can be expressed by

$$\Phi f(0) = \Phi f(1) = T(1), \qquad \Phi f(2) = T(2),$$

and

$$\Phi f(n) = h(n) + \frac{n-3}{n} f(n-1) + \frac{2\alpha}{(n-2)(n-1)n} \sum_{k=2}^{n-1}(k-1) f(k),$$

for all $n \geqslant 3$.

Observe that $\sum_{k=2}^{n-1}(k-1) = (n-1)(n-2)/2$. Then, for $g \in \mathcal{C}$ monotone increasing satisfying $T(n) \leqslant g(n)$, $n = 0, 1, 2$, and

$$\frac{c_1(4n-3) + 3c_2}{n} + \frac{n-3+\alpha}{n} g(n-1) \leqslant g(n), \quad n \geqslant 3, \tag{IV}$$

it follows that $\Phi g \leqslant g$, so Φ is an improver with respect to g, and hence, the solution f_T of the recurrence equation T verifies $f_T \leqslant g$ by Theorem 2.

Similarly to Cases (A)–(C), we show that for $0 < \alpha \leqslant 2$, $f_T(n) \in \mathcal{O}(n \log_a n)$, for any $a > 1$.

Indeed, note that for each $n>1$, we have $n + 1 > (n-3) + 4(n-2)/(n-1)$. Hence, it immediately follows from inequality $(**)$ that there is $n_0 \in \mathbb{N}$ such that for each $n \geqslant n_0$, and $0 < \alpha \leqslant 2$,

$$Kn \log_a n > \left(\frac{n-3}{n} + \frac{2\alpha(n-2)}{(n-1)n}\right) K(n-1)\log_a(n-1) + \frac{c_1(4n-3) + 3c_2}{n},$$

where $K > 4c_1 \cdot \ln a$.

So, inequality (IV) above is satisfied for $g(n) = Kn \log_a n$, with $K > 4c_1 \cdot \ln a$, and $n \geqslant n_0$.

We conclude that $f_T(n) \in \mathcal{O}(n \log_a n)$, whenever $0 < \alpha \leqslant 2$.

Acknowledgment

The authors are grateful to the referee for his suggestions.

References

[1] P. Flajolet, Analytic analysis of algorithms, in: W. Kuich (Ed.), 19th Internat. Colloq. ICALP'92, Vienna, July 1992; Automata, Languages and Programming, in: Lecture Notes in Comput. Sci., vol. 623, 1992, pp. 186–210.
[2] P. Fletcher, W.F. Lindgren, Quasi-Uniform Spaces, Marcel Dekker, New York, 1982.
[3] D.E. Knuth, The Art of Computer Programming, vol. 3: Sorting and Searching, Addison–Wesley, 1973.
[4] L.M. García-Raffi, S. Romaguera, E.A. Sánchez-Pérez, Sequence spaces and asymmetric norms in the theory of computational complexity, Math. Comput. Modelling 36 (2002) 1–11.
[5] H.P.A. Künzi, Nonsymmetric Topology, Bolyai Soc. Math. Stud., vol. 4, Topology, Szekszárd, 1993, Hungary, Budapest, 1995, pp. 303–338.
[6] H.P.A. Künzi, Nonsymmetric distances and their associated topologies: About the origins of basic ideas in the area of asymmetric topology, in: C.E. Aull, R. Lowen (Eds.), Handbook of the History of General Topology, vol. 3, Kluwer, Dordrecht, 2001, pp. 853–968.
[7] I.L. Reilly, P.V. Subrahmanyam, M.K. Vamanamurthy, Cauchy sequences in quasi-pseudo-metric spaces, Monatsh. Math. 93 (1982) 127–140.

[8] S. Romaguera, Left K-completeness in quasi-metric spaces, Math. Nachr. 157 (1992) 15–23.
[9] S. Romaguera, E.A. Sánchez-Pérez, O. Valero, The dual complexity space as the dual of a normed cone, Electron. Notes Theor. Comput. Sci. 161 (2006) 165–174.
[10] S. Romaguera, M. Schellekens, Quasi-metric properties of complexity spaces, Topology Appl. 98 (1999) 311–322.
[11] S. Romaguera, M. Schellekens, Duality and quasi-normability for complexity spaces, Appl. Gen. Topol. 3 (2002) 91–112.
[12] S. Romaguera, M. Schellekens, Partial metric monoids and semivaluation spaces, Topology Appl. 153 (2005) 948–962.
[13] M. Schellekens, The Smyth completion: A common foundation for denotational semantics and complexity analysis, in: Proc. MFPS 11, in: Electron. Notes Theor. Comput. Sci., vol. 1, 1995, pp. 535–556.
[14] D. Scott, C. Strachey, Toward a Mathematical Semantics for Computer Languages, Oxford Programming Research Group Technical Monograph, PRG-6, 1971.
[15] M.B. Smyth, Quasi-uniformities: Reconciling domains with metric spaces, in: M. Main, et al. (Eds.), Mathematical Foundations of Programming Language Semantics, 3rd Workshop, Tulane 1987, in: Lecture Notes in Comput. Sci., vol. 298, Springer, Berlin, 1988, pp. 236–253.
[16] M.B. Smyth, Totally bounded spaces and compact ordered spaces as domains of computation, in: G.M. Reed, A.W. Roscoe, R.F. Wachter (Eds.), Topology and Category Theory in Computer Science, Clarendon Press, Oxford, 1991, pp. 207–229.
[17] J.E. Stoy, Denotational Semantics: The Scott–Strachey Approach to Programming Language Theory, MIT Press, 1981.
[18] Ph. Sünderhauf, The Smyth-completion of a quasi-uniform space, in: M. Droste, Y. Gurevich (Eds.), Semantics of Programming Languages and Model Theory, in: Algebra Logic Appl., Gordon and Breach Sci. Publ., New York, 1993, pp. 189–212.