Theoretical Computer Science 401 (2008) 228–235

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

The complexity of Tarski's fixed point theorem[☆,☆☆]

Ching-Lueh Chang [a], Yuh-Dauh Lyuu [a,b,*], Yen-Wu Ti [a]

[a] *Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan*

[b] *Department of Finance, National Taiwan University, Taipei, Taiwan*

ARTICLE INFO

Article history:
Received 4 May 2007
Received in revised form 4 May 2008
Accepted 6 May 2008
Communicated by X. Deng

Keywords:
Tarski's fixed point theorem
Query complexity

ABSTRACT

Tarski's fixed point theorem guarantees the existence of a fixed point of an order-preserving function $f : L \to L$ defined on a nonempty complete lattice $(L, \preceq)$ [B. Knaster, Un théorème sur les fonctions d'ensembles, Annales de la Société Polonaise de Mathématique 6 (1928) 133–134; A. Tarski, A lattice theoretical fixpoint theorem and its applications, Pacific Journal of Mathematics 5 (1955) 285–309]. In this paper, we investigate several algorithmic and complexity-theoretic topics regarding Tarski's fixed point theorem. In particular, we design an algorithm that finds a fixed point of f when it is given $(L, \preceq)$ as input and f as an oracle. Our algorithm makes $O(\log |L|)$ queries to f when $\preceq$ is a total order on L. We also prove that when both f and $(L, \preceq)$ are given as oracles, any deterministic or randomized algorithm for finding a fixed point of f makes an expected $\Omega(|L|)$ queries for some $(L, \preceq)$ and f.

1. Introduction

A complete lattice is a set L endowed with a partial order $\preceq$ on L such that every nonempty subset of L has a greatest lower bound and a least upper bound in L. The complete lattice $(L, \preceq)$ is said to be nonempty if L is nonempty; it is said to be finite if L is finite. A function $f : L \to L$ is order-preserving if for every $x, y \in L$ with $x \preceq y$, we have $f(x) \preceq f(y)$. An element $x \in L$ satisfying $f(x) = x$ is called a fixed point of f. Given any nonempty complete lattice $(L, \preceq)$ and any order-preserving function $f : L \to L$, Tarski's fixed point theorem [2] states that the set of fixed points of f forms a nonempty complete lattice under the ordering induced by $\preceq$. The special case where L is a power set and $\preceq$ is the set inclusion relation is proven by Knaster [1] earlier, so Tarski's fixed point theorem is also known as the Knaster–Tarski theorem.

Tarski's fixed point theorem has been actively researched. Davis [3] proves a kind of converse of Tarski's fixed point theorem by showing that for any partially-ordered set $(L, \preceq)$ such that every order-preserving function $f : L \to L$ has a fixed point, it holds that $(L, \preceq)$ forms a complete lattice. Cousot and Cousot [4] and Echenique [5] give constructive versions of Tarski's fixed point theorem, and Cousot [6] gives asynchronous iterative methods for finding fixed points. Tarski's fixed point theorem has also found tremendous applications in abstract interpretation [7–9], subadditive and supermodular games [10–14], games with complementarities [15–19], stable matchings [20,21], monotone stopping games [22,23], production economies with income taxes [24] and iterated function systems [25].

This paper investigates algorithmic and complexity-theoretic issues of Tarski's fixed point theorem. Given a nonempty complete lattice $(L, \preceq)$ and oracle access to an order-preserving function $f : L \to L$, we are interested in the number of times needed to query f to find a fixed point. Despite extensive research on Tarski's fixed point theorem, the number of queries

☆ The authors are supported in part by NSC grant 96-2213-E-002-024.

☆☆ The authors thank the anonymous reviewers for their detailed comments.

* Corresponding author at: Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan.
E-mail addresses: d95007@csie.ntu.edu.tw (C.-L. Chang), lyuu@csie.ntu.edu.tw (Y.-D. Lyuu), d91010@csie.ntu.edu.tw (Y.-W. Ti).

0304-3975/$ – see front matter
doi:10.1016/j.tcs.2008.05.005

needed to find a fixed point of f remains open. Prior to our work, even for the degenerate case where $\preceq$ is a total order on a finite set L, the number of queries to $f : L \to L$ needed to find a fixed point has not been shown to be $o(|L|)$. In contrast, we propose an algorithm that needs to query f for only $O(\log |L|)$ times to find a fixed point of f when $\preceq$ is a total order on a finite set L. In general, the number of times that our algorithm queries f depends on several parameters concerning the lattice $(L, \preceq)$.

We also give hardness results on finding fixed points of an order-preserving function $f : L \to L$ on a finite complete lattice $(L, \preceq)$. When both the function and the partial order are given as oracles, we show that any randomized algorithm for finding a fixed point needs to query the oracles an expected $\Omega(|L|)$ times, for some nonempty finite complete lattice $(L, \preceq)$ and some order-preserving function $f : L \to L$. Thus, when both the function and the partial order are given as oracles, no randomized algorithm for a fixed point is asymptotically more efficient in terms of the expected number of queries than the one that simply tries all $x \in L$. Trivially, the same conclusion holds for deterministic algorithms.

Our paper is organized as follows. Section 2 gives basic definitions and facts. Section 3 presents our algorithm for finding fixed points. Section 4 establishes the lower bound on the number of queries for any algorithm that finds a fixed point. Section 5 concludes the paper.

2. Definitions

Let L be a set endowed with a partial order $\preceq$ [26]. For any $C \subseteq L$, $\max C$ (resp., $\min C$) denotes the element $x \in C$, if it exists, satisfying $y \preceq x$ (resp., $x \preceq y$) for every $y \in C$. Note that if $\max C$ (resp., $\min C$) exists, it is unique by antisymmetry [26] of the partial order $\preceq$. We say that $(L, \preceq)$ is a complete lattice if every nonempty set $B \subseteq L$ has a greatest lower bound and a least upper bound in L, i.e., $\max\{x \in L \mid \forall y \in B, x \preceq y\}$ and $\min\{x \in L \mid \forall y \in B, y \preceq x\}$ exist. In particular, by setting $B = L$, we see that $\min L$ and $\max L$ must exist for a complete lattice $(L, \preceq)$ with $L \neq \emptyset$ because $\max\{x \in L \mid \forall y \in L, x \preceq y\}$ and $\min\{x \in L \mid \forall y \in L, y \preceq x\}$ can exist only if $\{x \in L \mid \forall y \in L, x \preceq y\}$ and $\{x \in L \mid \forall y \in L, y \preceq x\}$ are nonempty. The restriction $f|_S : S \to L$ of a function $f : L \to L$ to a set $S \subseteq L$ means $f|_S(x) = f(x)$ for $x \in S$.

A function $f : L \to L$ is order-preserving if for every $x, y \in L$ with $x \preceq y$, we have $f(x) \preceq f(y)$. For $x, y \in L$, we write $x \prec y$ if $x \preceq y$ but $x \neq y$, and $x \not\preceq y$ if $x \preceq y$ does not hold. If at least one of $x \preceq y$ and $y \preceq x$ holds, x and y are said to be comparable with each other; otherwise, they are incomparable with each other. We denote by $f^i : L \to L$ the i-time repeated composition of f for $i \geq 1$. That is, $f^1 = f$ and $f^i = f \circ f^{i-1}$ for $i > 1$. By convention, $f^0 : L \to L$ is the identity function. For $x \in L$, if $f^i(x)$ is a fixed point of f for some $i \in \mathbb{N}$, we use $f^*(x)$ to denote that fixed point. If $f^i(x)$ is not a fixed point of f for any $i \in \mathbb{N}$, we say $f^*(x)$ does not exist. Note that $f^*(x)$ is well-defined whenever it exists because there are no $i_1, i_2 \in \mathbb{N}$ such that $f^{i_1}(x)$ and $f^{i_2}(x)$ are distinct fixed points of f.

The following is Tarski's fixed-point theorem [2].

Theorem 1 ([2]). *Let $(L, \preceq)$ be a nonempty complete lattice. For every order-preserving function $f : L \to L$, the set of fixed points of f with the ordering induced by $\preceq$ forms a nonempty complete lattice.*

In particular, Tarski's fixed-point theorem guarantees the existence of fixed points of any order-preserving function $f : L \to L$ on any nonempty complete lattice $(L, \preceq)$.

Definition 2. Let $(L, \preceq)$ be a complete lattice. A $\preceq$-chain $\boldsymbol{x}$ of length $k \in \mathbb{N}$ is a sequence $(x_1, \ldots, x_k)$ of elements in L such that $x_1 \preceq \cdots \preceq x_k$. We write $|\boldsymbol{x}| = k$ for the length of $\boldsymbol{x}$. The elements that appear on $\boldsymbol{x}$ are $x_1, \ldots, x_k$, and x_i appears before (resp. after) x_j for $i < j$ (resp. $i > j$). The element x_i is called the i-th component of $\boldsymbol{x}$. The $\preceq$-chain $(f(x_1), \ldots, f(x_k))$ is abbreviated as $f(\boldsymbol{x})$. The subchain of $\boldsymbol{x}$ from x_i to x_j, denoted $\boldsymbol{x}[x_i, x_j]$, is the $\preceq$-chain $(x_i, \ldots, x_j)$. Let $\boldsymbol{y} = (y_1, \ldots, y_h)$ be a $\preceq$-chain, $h \in \mathbb{N}$. We say that $\boldsymbol{x}$ is a superchain of $\boldsymbol{y}$ if each component of $\boldsymbol{y}$ appears on $\boldsymbol{x}$, and $\boldsymbol{x}$ is a proper superchain of $\boldsymbol{y}$ if we furthermore have $\boldsymbol{x} \neq \boldsymbol{y}$. If $|\boldsymbol{x}| = |\boldsymbol{y}|$, we write $\boldsymbol{x} \preceq \boldsymbol{y}$ if $x_i \preceq y_i$ for each $1 \leq i \leq k$. If $|\boldsymbol{x}| = |\boldsymbol{y}|$ but $\boldsymbol{x} \preceq \boldsymbol{y}$ does not hold, then we write $\boldsymbol{x} \not\preceq \boldsymbol{y}$. If $x_k \preceq y_1$, the concatenation $\boldsymbol{x} \circ \boldsymbol{y}$ of $\boldsymbol{x}$ and $\boldsymbol{y}$ is the $\preceq$-chain $(x_1, \ldots, x_k, y_1, \ldots, y_h)$. The number of fixed points of f appearing on $\boldsymbol{x}$ is denoted $|\boldsymbol{x}|_{\text{fixed}}$. We denote by $|\boldsymbol{x}|_{\text{incmp}}$ the number of $i \in \{1, \ldots, k\}$ such that x_i is incomparable with $f(x_i)$.

Definition 3. Let $(L, \preceq)$ be a complete lattice, $f : L \to L$ be order-preserving and $\boldsymbol{x} = (x_1, \ldots, x_k)$ be a $\preceq$-chain. If $x_i \neq x_j$ for $i \neq j$, then $\boldsymbol{x}$ is said to be a $\prec$-chain. A $\prec$-chain $\boldsymbol{y} = (y_1, \ldots, y_h)$ is maximal if there are no $z \in L$ and $0 \leq i \leq h$ such that

$$(y_1, \ldots, y_i, z, y_{i+1}, \ldots, y_h)$$

is a $\prec$-chain. The set of maximal $\prec$-chains of $(L, \preceq)$ is denoted $\mathcal{M}$. Finally, we define

$$\mathcal{X} \stackrel{\text{def}}{=} \{\boldsymbol{x} \in \mathcal{M} \mid |\boldsymbol{x}|_{\text{fixed}} = \max_{\boldsymbol{y} \in \mathcal{M}} |\boldsymbol{y}|_{\text{fixed}}\},$$

which is the set of maximal $\prec$-chains containing the largest number (over all maximal $\prec$-chains) of fixed points.

It is not hard to see that the first and last components of every maximal $\prec$-chain are $\min L$ and $\max L$, respectively. We will use the following easy lemma.

Lemma 4. *Let $(L, \preceq)$ be a nonempty finite complete lattice and $f : L \to L$ be order-preserving. If $x \in L$ is comparable with $f(x)$, then $f^*(x)$ exists.*

Proof. Assume without loss of generality that $x \preceq f(x)$, and therefore $f^i(x) \preceq f^{i+1}(x)$ for $i \in \mathbb{N}$. The finiteness of L implies that the sequence $\{f^i(x)\}_{i=0}^{\infty}$ contains repeated elements. But if $f^i(x) = f^j(x)$ for some $0 \le i < j$, then $f^i(x)$ is a fixed point of f by the fact that $f^i(x) \preceq f^{i+1}(x) \preceq \cdots \preceq f^j(x)$. □

An algorithm is given oracle access to a complete lattice $(L, \preceq)$ when it is able to query $(L, \preceq)$ on arbitrary $(x, y) \in L^2$. The answer to the query is either $x \preceq y$, $y \preceq x$ or none of the above. Similarly, an algorithm that is given oracle access to a function $f : L \to L$ may query its oracle on an arbitrary $x \in L$ to obtain $f(x)$.

3. An algorithm

Given a nonempty finite complete lattice $(L, \preceq)$ and oracle access to an order-preserving function $f : L \to L$, we are interested in the number of times needed to query f in order to find a fixed point. In this section, we find an upper bound on the query complexity as a function of several interesting parameters of $(L, \preceq)$. For this purpose, we design an algorithm which tries to find a fixed point of f on any given maximal $\prec$-chain via a binary search. The algorithm, called BIN-SEARCH, may fail to output a fixed point on a maximal $\prec$-chain even if fixed points do appear on it. Fortunately, we will show that BIN-SEARCH successfully finds a fixed point of f on at least one maximal $\prec$-chain. Then, we analyze the number of queries made by the algorithm, called FIXED-POINT later, that simply runs BIN-SEARCH over every maximal $\prec$-chain. The following lemmas will help derive conditions on a $\prec$-chain sufficient to guarantee BIN-SEARCH's successful discovery of a fixed point on it.

Lemma 5. *Let $(L, \preceq)$ be a nonempty finite complete lattice, $f : L \to L$ be order-preserving and $\boldsymbol{x} = (x_1, \ldots, x_k)$ be an arbitrary member of $\mathcal{X}$. For each $1 \le i \le k$ such that x_i is comparable with $f(x_i)$, the fixed point $f^*(x_i)$ appears on $\boldsymbol{x}$.*

Proof. Assume $i \in \{1, \ldots, k\}$ is such that x_i is comparable with $f(x_i)$. We shall only deal with the case that $x_i \preceq f(x_i)$; the case where $f(x_i) \preceq x_i$ is symmetric. Assume for contradiction that $f^*(x_i)$ does not appear on $\boldsymbol{x}$. Let $t \in \mathbb{N}$ be such that $f^0(x_i) \prec \cdots \prec f^t(x_i)$ are distinct and $f^t(x_i) = f^*(x_i)$ (note that this t exists by Lemma 4). We assume that $t > 0$, for otherwise the lemma trivially holds.

Suppose first that none of $x_{i+1}, \ldots, x_k$ is a fixed point. Then

$$\boldsymbol{y} = (x_1, \ldots, x_i, f(x_i), \ldots, f^t(x_i))$$

is a $\prec$-chain with $|\boldsymbol{y}|_{\text{fixed}} = |\boldsymbol{x}|_{\text{fixed}} + 1$. Hence any maximal $\prec$-chain that is a superchain of $\boldsymbol{y}$ has more fixed points than $\boldsymbol{x}$ does, a contradiction as $\boldsymbol{x} \in \mathcal{X}$.

Now suppose at least one of $x_{i+1}, \ldots, x_k$ is a fixed point and x_j is the one among them with the least index j. Since $x_i \prec x_j$, we must have $f^t(x_i) \preceq x_j$ by iteratively applying f for t times on x_i and x_j. But since $f^*(x_i) = f^t(x_i)$ does not appear on $\boldsymbol{x}$, we must have $f^t(x_i) \ne x_j$. Then

$$\boldsymbol{z} = (x_1, \ldots, x_i, f(x_i), \ldots, f^t(x_i), x_j, \ldots, x_k)$$

is a $\prec$-chain with $|\boldsymbol{z}|_{\text{fixed}} = |\boldsymbol{x}|_{\text{fixed}} + 1$. Any maximal $\prec$-chain that is a superchain of $\boldsymbol{z}$ has more fixed points than $\boldsymbol{x}$ does, again a contradiction as $\boldsymbol{x} \in \mathcal{X}$. □

The following lemma is easy to show.

Lemma 6. *Let $(L, \preceq)$ be a finite complete lattice and $f : L \to L$ be order-preserving. Let $x \in L$ be such that $f(x) \preceq x$ and $t \in \mathbb{N}$ be such that $f^t(x) \prec \cdots \prec f^0(x)$. Any superchain $\boldsymbol{z}$ of $(f^t(x), f^{t-1}(x), \ldots, f^0(x))$ satisfies $f(\boldsymbol{z}[f^t(x), f^0(x)]) \preceq \boldsymbol{z}[f^t(x), f^0(x)]$.*

Proof. Any element $y \in L$ that appears on $\boldsymbol{z}[f^t(x), f^0(x)]$ satisfies $f^{i+1}(x) \preceq y \preceq f^i(x)$ for some $0 \le i < t$. This implies $f^{i+2}(x) \preceq f(y) \preceq f^{i+1}(x)$ and therefore $f(y) \preceq f^{i+1}(x) \preceq y$. □

Lemma 7. *Let $(L, \preceq)$ be a finite complete lattice and $f : L \to L$ be order-preserving. Let $\boldsymbol{x} = (x_1, \ldots, x_k)$ be a member of $\mathcal{X}$ and $i \in \{1, \ldots, k\}$ be such that $f(x_i) \preceq x_i$. If $f(\boldsymbol{x}[f^*(x_i), x_i]) \not\preceq \boldsymbol{x}[f^*(x_i), x_i]$, then there exists a $\boldsymbol{z} \in \mathcal{X}$ satisfying $|\boldsymbol{z}|_{incmp} < |\boldsymbol{x}|_{incmp}$.*

Proof. That $f^*(x_i)$ exists and appears on $\boldsymbol{x}$ is guaranteed by Lemma 5. If $x_i = f^*(x_i)$, then the lemma holds because $f(x_i) \preceq x_i$ and the only element appearing on $\boldsymbol{x}[f^*(x_i), x_i]$ is x_i, invalidating the premise $f(\boldsymbol{x}[f^*(x_i), x_i]) \not\preceq \boldsymbol{x}[f^*(x_i), x_i]$. So we assume otherwise. We first argue that no fixed point of f except $f^*(x_i)$ appears on $\boldsymbol{x}[f^*(x_i), x_i]$. This is because if $q \in L$ is a fixed point with $f^*(x_i) \prec q \preceq x_i$, then

$$q = f^*(q) \preceq f^*(x_i)$$

by iteratively applying f on q and x_i, a contradiction.

Assume that

$$\boldsymbol{x}[f^*(x_i), x_i] = (f^*(x_i), \ldots, x_h, \ldots, x_i)$$

where $f(x_h) \not\preceq x_h$. If no such $x_h \in L$ exists, the lemma trivially holds. We are left with two possibilities: either $x_h \prec f(x_h)$ or x_h is incomparable with $f(x_h)$.

Set $i \leftarrow 0, a_0 \leftarrow x_1, b_0 \leftarrow x_k$ and $c_0 \leftarrow x_{\lceil (k+1)/2 \rceil}$;
repeat
  if $f(c_i) \preceq c_i$ **then**
    $a_{i+1} \leftarrow a_i$ and $b_{i+1} \leftarrow c_i$;
  else
    $a_{i+1} \leftarrow c_i$ and $b_{i+1} \leftarrow b_i$;
  end if
  Find $A \in \{1, \ldots, k\}$ satisfying $a_{i+1} = x_A$;
  Find $B \in \{1, \ldots, k\}$ satisfying $b_{i+1} = x_B$;
  $c_{i+1} \leftarrow x_{\lceil \frac{A+B}{2} \rceil}$
  $i \leftarrow i + 1$;
until $a_i = b_i$ or c_i is a fixed point of f
if c_i is a fixed point of f **then**
  Output c_i;
end if

Fig. 1. Algorithm BIN-SEARCH is given as input a nonempty finite complete lattice $(L, \preceq)$ and a maximal $\prec$-chain $\boldsymbol{x} = (x_1, \ldots, x_k)$. It is also given oracle access to an order-preserving function $f : L \to L$.

Suppose that $x_h \prec f(x_h)$. Lemma 5 guarantees that $f^*(x_h)$ exists and appears on $\boldsymbol{x}$. Also, since $f^*(x_i) \prec x_h \preceq x_i$ ($f^*(x_i) \neq x_h$ because otherwise $x_h = f(x_h)$), we must have $f^*(x_i) \preceq f^*(x_h) \preceq f^*(x_i)$ by iteratively applying f, implying $f^*(x_h) = f^*(x_i) \prec x_h$. But the assumption that $x_h \prec f(x_h)$ implies $x_h \preceq f(x_h) \preceq \cdots \preceq f^*(x_h)$, a contradiction.

Now suppose that x_h is incomparable with $f(x_h)$. Let $t \in \mathbb{N}$ be such that $f^t(x_i) \prec \cdots \prec f^0(x_i)$ are distinct and $f^t(x_i) = f^*(x_i)$. Lemma 5 guarantees the existence of a $1 \leq j \leq k$ such that $x_j = f^*(x_i)$. Write $\boldsymbol{u} = (x_1, \ldots, x_{j-1})$, $\boldsymbol{v} = (f^t(x_i), \ldots, f^0(x_i))$ and $\boldsymbol{w} = (x_{i+1}, \ldots, x_k)$. Since we have seen that no fixed point of f except $f^t(x_i) = f^*(x_i)$ appears on $\boldsymbol{x}[f^*(x_i), x_i] = \boldsymbol{x}[x_j, x_i]$, the $\prec$-chain

$$\boldsymbol{u} \circ \boldsymbol{v} \circ \boldsymbol{w} = (x_1, \ldots, x_{j-1}, x_j = f^t(x_i), f^{t-1}(x_i), \ldots, x_i = f^0(x_i), x_{i+1}, \ldots, x_k)$$

satisfies $|\boldsymbol{u} \circ \boldsymbol{v} \circ \boldsymbol{w}|_{\text{fixed}} = |\boldsymbol{x}|_{\text{fixed}}$. Let $\boldsymbol{z}$ be any superchain of $\boldsymbol{u} \circ \boldsymbol{v} \circ \boldsymbol{w}$ that is a maximal $\prec$-chain. It is clear that $\boldsymbol{z} \in \mathcal{X}$ since

$$|\boldsymbol{z}|_{\text{fixed}} \geq |\boldsymbol{u} \circ \boldsymbol{v} \circ \boldsymbol{w}|_{\text{fixed}} = |\boldsymbol{x}|_{\text{fixed}}.$$

Furthermore, since $\boldsymbol{x} = (x_1, \ldots, x_k)$ is a maximal $\prec$-chain, there is no proper superchain of $(x_1, \ldots, x_j)$ whose last component is x_j. Similarly, there is no proper superchain of $(x_i, \ldots, x_k)$ with x_i being the first component. Consequently, we can write $\boldsymbol{z} = \boldsymbol{u} \circ \boldsymbol{v}' \circ \boldsymbol{w}$ where $\boldsymbol{v}'$ is a superchain of $\boldsymbol{v}$ and the first (resp., last) component of $\boldsymbol{v}'$ is $f^t(x_i)$ (resp., $f^0(x_i)$). Lemma 6 shows that $f(\boldsymbol{v}'[f^t(x_i), f^0(x_i)]) \preceq \boldsymbol{v}'[f^t(x_i), f^0(x_i)]$ and therefore $|\boldsymbol{v}'|_{\text{incmp}} = 0$. Thus,

$$|\boldsymbol{z}|_{\text{incmp}} = |\boldsymbol{u}|_{\text{incmp}} + |\boldsymbol{w}|_{\text{incmp}} < |\boldsymbol{u}|_{\text{incmp}} + |\boldsymbol{w}|_{\text{incmp}} + \left|\boldsymbol{x}[x_j, x_i]\right|_{\text{incmp}} = |\boldsymbol{x}|_{\text{incmp}}.$$

The inequality holds because we have assumed that x_h is incomparable with $f(x_h)$ and x_h appears on $\boldsymbol{x}[f^*(x_i), x_i] = \boldsymbol{x}[x_j, x_i]$. $\square$

Given a nonempty finite complete lattice $(L, \preceq)$, a maximal $\prec$-chain $\boldsymbol{x}$ and oracle access to an order-preserving function $f : L \to L$, the algorithm BIN-SEARCH in Fig. 1 tries to find a fixed point of f on $\boldsymbol{x}$ via binary search. Although BIN-SEARCH is not guaranteed to find a fixed point of f on every maximal $\prec$-chain on which a fixed point appears, the following theorem shows that BIN-SEARCH successfully finds a fixed point of f on at least one maximal $\prec$-chain. By summing up the number of queries needed to run BIN-SEARCH over every maximal $\prec$-chain, we derive an upper bound on the number of queries needed to find a fixed point of f.

Theorem 8. *There is an algorithm (call it FIXED-POINT) that, given a nonempty finite complete lattice $(L, \preceq)$ and oracle access to an order-preserving function $f : L \to L$, outputs a fixed point of f. The number of times that FIXED-POINT queries f is*

$$O(|\mathcal{M}| \log \max_{\boldsymbol{x} \in \mathcal{M}} |\boldsymbol{x}|).$$

Proof. Let $\boldsymbol{x} = (x_1, \ldots, x_k)$ be any member of $\mathcal{X}$ satisfying $|\boldsymbol{x}|_{\text{incmp}} = \min_{\boldsymbol{y} \in \mathcal{X}} |\boldsymbol{y}|_{\text{incmp}}$. Lemmas 5 and 7 imply that $\boldsymbol{x}$ satisfies the following two properties.

1. If $y \in L$ appears on $\boldsymbol{x}$ and y is comparable with $f(y)$, then $f^*(y)$ also appears on $\boldsymbol{x}$.
2. If $y \in L$ appears on $\boldsymbol{x}$ and $f(y) \preceq y$, then $f(\boldsymbol{x}[f^*(y), y]) \preceq \boldsymbol{x}[f^*(y), y]$.

Consider running BIN-SEARCH with input $(L, \preceq)$, $\boldsymbol{x}$ and oracle access to f. We now show by induction on $i \in \mathbb{N}$ that there exists a fixed point q_i of f that appears on $\boldsymbol{x}[a_i, b_i]$ and satisfies $f(\boldsymbol{x}[q_i, b_i]) \preceq \boldsymbol{x}[q_i, b_i]$, for every $i \geq 0$. For $i = 0$, we must have $a_0 = x_1 = \min L$ and $b_0 = x_k = \max L$ since $\boldsymbol{x}$ is a maximal $\prec$-chain. This and properties 1–2 above imply that $f^*(\max L)$ appears on $\boldsymbol{x}$ and

$$f\left(\boldsymbol{x}\left[f^*(\max L), \max L\right]\right) \preceq \boldsymbol{x}\left[f^*(\max L), \max L\right],$$

establishing the base of the induction.

Assume as induction hypothesis that a fixed point q_i of f appears on $\boldsymbol{x}[a_i, b_i]$ and

$$f(\boldsymbol{x}[q_i, b_i]) \preceq \boldsymbol{x}[q_i, b_i]. \tag{1}$$

We are to prove the existence of a fixed point q_{i+1} of f that appears on $\boldsymbol{x}[a_{i+1}, b_{i+1}]$ and satisfies

$$f(\boldsymbol{x}[q_{i+1}, b_{i+1}]) \preceq \boldsymbol{x}[q_{i+1}, b_{i+1}],$$

assuming BIN-SEARCH does not exit after the i-th iteration. For this purpose, we separate the discussion according to whether c_i appears before q_i on $\boldsymbol{x}[a_i, b_i]$.

Suppose first that c_i does not appear before q_i on $\boldsymbol{x}[a_i, b_i]$. Then Eq. (1) implies that $f(\boldsymbol{x}[q_i, c_i]) \preceq \boldsymbol{x}[q_i, c_i]$ and thus $\boldsymbol{x}[a_{i+1}, b_{i+1}] = \boldsymbol{x}[a_i, c_i]$. Therefore, q_i appears on $\boldsymbol{x}[a_{i+1}, b_{i+1}]$ and $f(\boldsymbol{x}[q_i, b_{i+1}]) \preceq \boldsymbol{x}[q_i, b_{i+1}]$ holds, completing the induction step.

Now suppose that c_i appears before q_i on $\boldsymbol{x}$. If $f(c_i) \not\preceq c_i$, then $\boldsymbol{x}[a_{i+1}, b_{i+1}] = \boldsymbol{x}[c_i, b_i]$ and the induction step follows from Eq. (1). If $f(c_i) \preceq c_i$, then $\boldsymbol{x}[a_{i+1}, b_{i+1}] = \boldsymbol{x}[a_i, c_i]$. But properties 1–2 imply that $f^*(c_i)$ appears on $\boldsymbol{x}$ and $f(\boldsymbol{x}[f^*(c_i), c_i]) \preceq \boldsymbol{x}[f^*(c_i), c_i]$. Hence, to complete the the induction step, we need only prove that $f^*(c_i)$ appears on $\boldsymbol{x}[a_i, c_i]$. This could be false only if $f^*(c_i)$ appears before a_i, which implies that there must have been some $j < i$ during the execution of BIN-SEARCH such that $f^*(c_i)$ does not appear before a_j but appears before a_{j+1} (note that $f^*(c_i)$ cannot appear before a_0). Consequently, we have $a_j \neq a_{j+1}$ and therefore

$$f(c_j) \not\preceq c_j \tag{2}$$

in step 3 of BIN-SEARCH and $a_{j+1} = c_j$ in step 6, which implies

$$c_j = a_{j+1} \preceq c_i \tag{3}$$

(note that c_i appears on each of $\boldsymbol{x}[a_0, b_0], \dots, \boldsymbol{x}[a_{j+1}, b_{j+1}], \dots, \boldsymbol{x}[a_i, b_i]$). Since we assume $f^*(c_i)$ appears before a_{j+1},

$$f^*(c_i) \prec a_{j+1}. \tag{4}$$

Eqs. (3) and (4) imply

$$f^*(c_i) \prec c_j \preceq c_i. \tag{5}$$

Property 2 implies

$$f(\boldsymbol{x}[f^*(c_i), c_i]) \preceq \boldsymbol{x}[f^*(c_i), c_i],$$

which together with Eq. (5) shows

$$f(c_j) \preceq c_j,$$

contradicting Eq. (2).

At this point we have shown the existence of a fixed point q_i of f that appears on $\boldsymbol{x}[a_i, b_i]$ and satisfies $f(\boldsymbol{x}[q_i, b_i]) \preceq \boldsymbol{x}[q_i, b_i]$, $i \geq 0$. This implies that when $\boldsymbol{x}$ is fed to BIN-SEARCH, any time the range of search narrows down from $\boldsymbol{x}[a_i, b_i]$ to $\boldsymbol{x}[a_{i+1}, b_{i+1}]$, at least one fixed point still appears on $\boldsymbol{x}[a_{i+1}, b_{i+1}]$. The algorithm FIXED-POINT simply feeds each maximal $\prec$-chain to BIN-SEARCH and return any fixed point found. The query complexity follows. □

In Theorem 8, the number of queries FIXED-POINT makes depends on the parameters $|\mathcal{M}|$ and $\max_{\boldsymbol{x}\in\mathcal{M}} |\boldsymbol{x}|$ of the lattice $(L, \preceq)$. But it is independent of the function f. The following fact is well-known (see, e.g., [27,5]).

Fact 1. *Let $(L, \preceq)$ be a nonempty complete lattice and $f : L \to L$ be order-preserving. If L is finite, $f^*(\max L)$ and $f^*(\min L)$ exist. If L is the infinite complete lattice $[\,0, 1\,]$ with the ordinary ordering on real numbers and f is continuous, then the sequences $\{f^n(0)\}_{n=0}^{\infty}$ and $\{f^n(1)\}_{n=0}^{\infty}$ are increasing and decreasing, respectively, and both converge to fixed points of f.*

Fact 1 implies that to find a fixed point of an order-preserving function $f : L \to L$ on a nonempty finite complete lattice, one needs only iteratively compute $f^n(\max L)$ (or $f^n(\min L)$) for $n = 0, 1, \dots$ until a fixed point is reached. However, even if $\preceq$ is restricted to be a total order on L, this iterative method needs to evaluate f at least $\Omega(|L|)$ times in the worst case. To see this, consider $L = \{1, \dots, |L|\}$, $\preceq$ being the ordinary ordering on natural numbers, $f(x) = x + 1$ for $x < \lceil |L|/2 \rceil$, $f(x) = x - 1$ for $x > \lceil |L|/2 \rceil$ and $f(\lceil |L|/2 \rceil) = \lceil |L|/2 \rceil$. In contrast, Theorem 8 reduces this $\Omega(|L|)$ query complexity to only $O(\log |L|)$ when $\preceq$ is a total order on L.

Theorem 8 also leads to an efficient method for finding an approximate fixed point of an order-preserving function $f : [\,0, 1\,] \to [\,0, 1\,]$. Finding an approximate fixed point of f means finding an $x^* \in [\,0, 1\,]$ with $|f(x^*) - x^*| < \epsilon$ for a given $\epsilon > 0$.

Theorem 9. *There is an algorithm that, given $\epsilon > 0$ and oracle access to an order-preserving function $f : [0, 1] \to [0, 1]$, outputs an $x^* \in [0, 1]$ satisfying $|f(x^*) - x^*| < \epsilon$ after making $O(\log(1/\epsilon))$ queries to f.*

Proof. Let $n = \lceil 1/\epsilon \rceil$ and $S = \{0/n, 1/n, \ldots, n/n\}$. Assume $g : S \to S$ maps each $x \in S$ to the element in S which is closest to $f(x)$, breaking ties by favoring smaller values in S. Clearly, $g(x) \le g(x')$ whenever $f(x) \le f(x')$, for $x, x' \in [0, 1]$. This and the fact that f is order-preserving imply that g is also order-preserving. Applying Theorem 8 to g, we see that a fixed point x^* of g can be found with $O(\log|S|) = O(\log(1/\epsilon))$ queries to g, or more clearly, $O(\log(1/\epsilon))$ queries to f because each query to g can be resolved by one query to f. But the fixed point x^* of g must satisfy $|f(x^*) - x^*| \le 1/(2n) < \epsilon$ by the definitions of S and g. $\square$

Given an order-preserving, continuous function $f : [0, 1] \to [0, 1]$, Fact 1 implies that $|f(f^n(0)) - f^n(0)| < \epsilon$ and $|f(f^n(1)) - f^n(1)| < \epsilon$ hold for sufficiently large $n \in \mathbb{N}$. Hence, one may find an $x^* \in [0, 1]$ satisfying $|f(x^*) - x^*| < \epsilon$ by iteratively computing $f^n(0)$ (or $f^n(1)$) for $n = 0, 1, \ldots$ until such an x^* is found. But this iterative method requires $\Omega(1/\epsilon)$ queries to f for some order-preserving, continuous $f : [0, 1] \to [0, 1]$. To see this, define $f : [0, 1] \to [0, 1]$ by $f(x) = x + \epsilon$ for $x \in [0, 1/2 - \epsilon]$, $f(x) = 1/2$ for $x \in (1/2 - \epsilon, 1/2 + \epsilon)$ and $f(x) = x - \epsilon$ for $x \in (1/2 + \epsilon, 1]$. For this f and $0 \le i \le 1/(2\epsilon) - 1$, we have $f^i(0) = \epsilon i$ and $f^i(1) = 1 - \epsilon i$. So neither $f^i(0)$ nor $f^i(1)$ equals the unique fixed point $1/2$ of f for $0 \le i \le 1/(2\epsilon) - 1$. In contrast, Theorem 9 shows an exponential speedup from $\Omega(1/\epsilon)$ to $O(\log 1/\epsilon)$ for finding approximate fixed points, and it does not require the continuity of f.

4. Evasive queries

In this section, we show that given as oracles a nonempty finite complete lattice $(L, \preceq)$ and an order-preserving function $f : L \to L$, the problem of finding a fixed point is evasive for all (deterministic or randomized) algorithms in the sense that any algorithm for a fixed point makes an expected $\Omega(|L|)$ queries for some $(L, \preceq)$ and f.

Theorem 10. *Let $(L, \preceq)$ be a nonempty finite complete lattice and $f : L \to L$ be an order-preserving function. Given oracle access to $(L, \preceq)$ and f, any randomized algorithm that finds a fixed point of f must make an expected $\Omega(|L|)$ queries to its oracles for some $(L, \preceq)$ and f.*

Proof. Let $L' = \{0, \ldots, n - 3\}$ and $L = L' \cup \{-\infty, \infty\}$ for an integer $n > 3$. Define $\preceq$ by

$$-\infty \preceq \ell \preceq \ell \preceq \infty$$

for $\ell \in L$, and

$$\ell_1 \not\preceq \ell_2$$

for all distinct $\ell_1, \ell_2 \in L \setminus \{-\infty, \infty\}$. That is, $-\infty$ and ∞ are the minimum and maximum elements under the ordering $\preceq$, respectively, and all other elements are incomparable with each other under $\preceq$. Define $f : L \to L$ by

$$f(-\infty) = -\infty,$$
$$f(\infty) = \infty$$

and

$$f(x) = x + 1 \bmod (n - 2)$$

for $x \in L \setminus \{-\infty, \infty\}$. Note that $-\infty$ and ∞ are the only fixed points of f.

Let $\boldsymbol{\pi} : L \to L$ be a random permutation over L in the sense that each of the $n!$ permutations is picked with equal probability. Define the relation $\preceq^{\boldsymbol{\pi}}$ so that $x \preceq^{\boldsymbol{\pi}} y$ if and only if $\boldsymbol{\pi}(x) \preceq \boldsymbol{\pi}(y)$. Set $f^{\boldsymbol{\pi}}(x) = \boldsymbol{\pi}^{-1}(f(\boldsymbol{\pi}(x)))$ for $x \in L$. It can be easily verified that $(L, \preceq^{\boldsymbol{\pi}})$ is a complete lattice, $f^{\boldsymbol{\pi}}$ is order-preserving and the fixed points of $f^{\boldsymbol{\pi}}$ are $\boldsymbol{\pi}^{-1}(-\infty)$ and $\boldsymbol{\pi}^{-1}(\infty)$. Note that if $x, y \in L$ are such that $\boldsymbol{\pi}(x), \boldsymbol{\pi}(y) \notin \{-\infty, \infty\}$ and $\boldsymbol{\pi}(y) = \boldsymbol{\pi}(x) + 1 \bmod (n - 2)$, then $y = f^{\boldsymbol{\pi}}(x)$.

Now consider feeding an arbitrary randomized algorithm ALG that finds a fixed point of $f^{\boldsymbol{\pi}}$ with oracle access to $(L, \preceq^{\boldsymbol{\pi}})$ and $f^{\boldsymbol{\pi}}$. For the purpose of proving lower bounds on the number of queries of ALG, we may replace the oracles provided to ALG by $\boldsymbol{\pi}$ and $\boldsymbol{\pi}^{-1}|_{L'}$. This is explained below.

(a) Each query $(x, y) \in L^2$ to $(L, \preceq^{\boldsymbol{\pi}})$ can be resolved by querying x and y to $\boldsymbol{\pi}$ and using the answers $\boldsymbol{\pi}(x)$ and $\boldsymbol{\pi}(y)$ to determine whether $x \preceq^{\boldsymbol{\pi}} y$, $y \preceq^{\boldsymbol{\pi}} x$ or none of the above holds.
(b) Each query $x \in L$ to $f^{\boldsymbol{\pi}}$ can be replaced by the query x to $\boldsymbol{\pi}$, followed by the query $\boldsymbol{\pi}(x) + 1 \bmod (n-2)$ to $\boldsymbol{\pi}^{-1}|_{L'}$ if $\boldsymbol{\pi}(x) \notin \{-\infty, \infty\}$. The original query x to $f^{\boldsymbol{\pi}}$ is answered as x itself if $\boldsymbol{\pi}(x) \in \{-\infty, \infty\}$, and as $\boldsymbol{\pi}^{-1}|_{L'}(\boldsymbol{\pi}(x) + 1 \bmod (n - 2))$ otherwise.

Here we do not feed the unrestricted mapping $\boldsymbol{\pi}^{-1}$ as an oracle to ALG, for otherwise the fixed point $\boldsymbol{\pi}^{-1}(\infty)$ can be obtained in one query, giving ALG excessively strong capability for finding fixed points. We may also assume without loss of generality that the last query of ALG is a fixed point of $f^{\boldsymbol{\pi}}$ and is asked to $\boldsymbol{\pi}$; this increases the query complexity of ALG by at most one, a constant.

For $t \geq 0$, denote by $Q(\boldsymbol{\pi}, t)$ and $Q(\boldsymbol{\pi}^{-1}|_{L'}, t)$ the sets of queries asked to $\boldsymbol{\pi}$ and $\boldsymbol{\pi}^{-1}|_{L'}$, respectively, among the first t queries of ALG. Define

$$S_t \stackrel{\text{def}}{=} Q(\boldsymbol{\pi}, t) \cup \boldsymbol{\pi}^{-1}|_{L'}\left(Q(\boldsymbol{\pi}^{-1}|_{L'}, t)\right)$$

to be the set of $x \in L$ whose $\boldsymbol{\pi}$-value is revealed among the first t queries, either because x is asked to $\boldsymbol{\pi}$ or because $\boldsymbol{\pi}(x) \in L'$ is asked to $\boldsymbol{\pi}^{-1}|_{L'}$. As can be easily verified, the restricted mapping $\boldsymbol{\pi}|_{S_t} : S_t \to \boldsymbol{\pi}(S_t)$ is completely known from the results of the first t queries, and no member of $L \setminus S_t$ and $L \setminus \boldsymbol{\pi}(S_t)$ is ever queried among the first t queries to $\boldsymbol{\pi}$ and $\boldsymbol{\pi}^{-1}$, respectively. Therefore, conditioned on any particular realization of S_t and $\boldsymbol{\pi}|_{S_t}$, the restricted mapping $\boldsymbol{\pi}|_{L\setminus S_t}$ is a random permutation from $L \setminus S_t$ to $L \setminus \boldsymbol{\pi}(S_t)$. This implies that conditioned on any particular realization of S_t and $\boldsymbol{\pi}|_{S_t}$ satisfying $\boldsymbol{\pi}(S_t) \cap \{-\infty, \infty\} = \emptyset$, the probability of $\boldsymbol{\pi}(S_{t+1}) \cap \{-\infty, \infty\} \neq \emptyset$ is

$$\frac{|\{-\infty, \infty\}|}{|L \setminus \boldsymbol{\pi}(S_t)|} = \frac{2}{n - |S_t|}$$

if the $(t+1)$-th query q_{t+1} is a member of $L \setminus S_t$ and is asked to $\boldsymbol{\pi}$. In case $q_{t+1} \in S_t$ is asked to $\boldsymbol{\pi}$ or $q_{t+1} \in L'$ is asked to $\boldsymbol{\pi}^{-1}|_{L'}$, the above conditional probability of $\boldsymbol{\pi}(S_{t+1}) \cap \{-\infty, \infty\} \neq \emptyset$ is zero. Now since $|S_t| \leq t$ for $t \geq 0$,

$$\Pr[\,\boldsymbol{\pi}(S_{t+1}) \cap \{-\infty, \infty\} \neq \emptyset \mid \boldsymbol{\pi}(S_t) \cap \{-\infty, \infty\} = \emptyset\,] \leq \frac{2}{n-t}, \tag{6}$$

where the probability is taken over $\boldsymbol{\pi}$ and the random coin tosses of ALG.

Denote by E_t the event $\boldsymbol{\pi}(S_t) \cap \{-\infty, \infty\} \neq \emptyset$, $t \geq 0$. For $0 \leq T \leq n$,

$$\begin{aligned}
&\Pr\left[\text{ ALG queries a fixed point of } f^{\boldsymbol{\pi}} \text{ to } \boldsymbol{\pi} \text{ within } T \text{ queries }\right] \\
&\leq \Pr[\,E_t \text{ for some } 1 \leq t \leq T\,] \\
&\leq \sum_{t=0}^{T-1} \Pr[\,E_{t+1} \text{ and } \neg E_t\,] \\
&\leq \sum_{t=0}^{T-1} \Pr[\,E_{t+1} \mid \neg E_t\,] \\
&\leq \sum_{t=0}^{T-1} \frac{2}{n-t},
\end{aligned}$$

where the second inequality follows by the fact that $\neg E_0$ and the last inequality by Eq. (6). This and our assumption that the last query of ALG is a fixed point of $f^{\boldsymbol{\pi}}$ asked to $\boldsymbol{\pi}$ imply

$$\Pr[\text{ ALG makes no more than } T \text{ queries }] \leq \sum_{t=0}^{T-1} \frac{2}{n-t}, \tag{7}$$

where the probability is taken over $\boldsymbol{\pi}$ and the random coin tosses of ALG.

By taking an averaging argument over all permutations $\pi : L \to L$ in Eq. (7), we see that there exists a permutation $\pi : L \to L$ such that given oracle access to $(L, \preceq^{\pi})$ and f^{π} (or in fact $\boldsymbol{\pi}$ and $\boldsymbol{\pi}^{-1}|_{L'}$ as explained earlier),

$$\Pr[\text{ ALG makes no more than } T \text{ queries }] \leq \sum_{t=0}^{T-1} \frac{2}{n-t}, \tag{8}$$

where the probability is taken over the random coin tosses of ALG. The expected number of queries of ALG is thus at least $(T+1)(1 - \sum_{t=0}^{T-1} 2/(n-t))$, which is $\Omega(n)$ for $T = 0.01 \cdot n$. □

The following corollary for deterministic algorithms is immediate.

Corollary 11. *Let $(L, \preceq)$ be a nonempty finite complete lattice and $f : L \to L$ be an order-preserving function. Given oracle access to $(L, \preceq)$ and f, any deterministic algorithm that finds a fixed point of f must make $\Omega(|L|)$ queries to its oracles for some $(L, \preceq)$ and f.*

5. Conclusion

Tarski's fixed point theorem and its applications have been extensively researched. However, the complexity of finding a fixed point is not yet well-known. To the best of our knowledge, it is open whether $o(|L|)$ queries to f are sufficient to find a fixed point of an order-preserving function $f : L \to L$ defined on a nonempty complete lattice $(L, \preceq)$. Our algorithm achieves this $o(|L|)$ bound on some lattices and leaves open the problem on general lattices. When both the function and the partial order are given as oracles, our lower bound eliminates the possibility of achieving an expected $o(|L|)$ query complexity for randomized algorithms. It is future work to investigate the query complexity for special classes of complete lattices and to explore how the query complexity changes correspondingly as to whether the partial order is provided as an oracle or an input.

References

[1] B. Knaster, Un théorème sur les fonctions d'ensembles, Annales de la Société Polonaise de Mathématique 6 (1928) 133–134.
[2] A. Tarski, A lattice theoretical fixpoint theorem and its applications, Pacific Journal of Mathematics 5 (1955) 285–309.
[3] A.C. Davis, A characterization of complete lattices, Pacific Journal of Mathematics 5 (1955) 311–319.
[4] P. Cousot, R. Cousot, Constructive versions of Tarski's fixed point theorems, Pacific Journal of Mathematics 82 (1) (1979) 43–57.
[5] F. Echenique, A short and constructive proof of Tarski's fixed-point theorem, International Journal of Game Theory 33 (2) (2005) 215–218.
[6] P. Cousot, Asynchronous iterative methods for solving a fixed point system of monotone equations in a complete lattice, Tech. Rep. R.R. 88, Laboratoire IMAG, Université scientifique et médicale de Grenoble, 1977.
[7] P. Cousot, R. Cousot, Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints, in: Proceedings of the 4th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages, 1977, pp. 238–252.
[8] P. Cousot, R. Cousot, Abstract interpretation and application to logic programs, Journal of Logic Programming 13 (2 & 3) (1992) 103–179.
[9] P. Cousot, R. Cousot, Refining model checking by abstract interpretation, Automated Software Engineering: An International Journal 6 (1) (1999) 69–95.
[10] D.M. Topkis, Equilibrium points in nonzero-sum n-person subadditive games, Tech. Rep. ORC 70-38, Operations Research Center, University of California, Berkeley, 1970.
[11] D.M. Topkis, Equilibrium points in nonzero-sum n-person submodular games, SIAM Journal on Control and Optimization 17 (1979) 773–787.
[12] P. Milgrom, J. Roberts, Comparing equilibria, American Economic Review 84 (3) (1994) 441–459.
[13] V. d'Orey, Fixed point theorems for correspondences with values in a partially ordered set and extended supermodular games, Journal of Mathematical Economics 25 (1996) 345–354.
[14] T.V. Zandt, Interim Bayesian Nash equilibrium on universal type spaces for supermodular games, Tech. Rep. 2007/14/EPS, INSEAD Business School, 2007.
[15] D.M. Topkis, Minimizing a submodular function on a lattice, Operations Research 26 (2) (1978) 305–321.
[16] X. Vives, Nash equilibrium with strategic complementarities, Journal of Mathematical Economics 19 (1990) 305–321.
[17] D.M. Topkis, Supermodularity and Complementarity, Princeton University Press, 1998.
[18] X. Vives, Oligopoly Pricing, MIT Press, 1999.
[19] F. Echenique, The equilibrium set of two-player games with complementarities is a sublattice, Economic Theory 22 (4) (2003) 903–905.
[20] H. Adachi, A search-theoretic approach to two-sided matching, Ph.D. Thesis, SUNY-Buffalo, 1998.
[21] H. Adachi, On a characterization of stable matchings, Economics Letters 68 (2000) 43–49.
[22] J.M. Mamer, Monotone stopping games, Journal of Applied Probability 24 (2) (1987) 386–401.
[23] Y. Ohtsubo, On a discrete-time non-zero-sum Dynkin problem with monotonicity, Journal of Applied Probability 28 (2) (1991) 466–472.
[24] W.J. Coleman II, Equilibrium in a production economy with an income tax, Econometrica 59 (4) (1991) 1091–1104.
[25] J. Jachymski, L. Gajek, K. Pokarowski, The Tarski-Kantorovitch principle and the theory of iterated function systems, Bulletin of the Australian Mathematical Society 61 (2) (2000) 247–261.
[26] R.P. Grimaldi, Discrete and Combinatorial Mathematics: An Applied Introduction, Addison-Wesley Longman Publishing Co., Inc., 1998.
[27] A. Granas, J. Dugundji, Fixed Point Theory, Springer-Verlag, 2003.