Upper bounds for stabilization in acyclic preference-based systems

Fabien Mathieu

Upper Bounds for Stabilization
in Acyclic Preference-Based Systems

Fabien Mathieu

Orange Labs, 38-40 rue du général Leclerc, 92794 Issy-les-Moulineaux, France
fabien.mathieu@orange-ftgroup.com
http://gyroweb.inria.fr/~fmathieu/

Abstract. Preference-based systems (p.b.s.) describe interactions between nodes of a system that can rank their neighbors. Previous work has shown that p.b.s. converge to a unique locally stable matching if an *acyclicity* property is verified. In the following we analyze acyclic p.b.s. with respect to the self-stabilization theory. We prove that the round complexity is bounded by $\frac{n}{2}$ for the adversarial daemon. The step complexity is equivalent to $\frac{n^2}{4}$ for the round robin daemon, and exponential for the general adversarial daemon.

Keywords: Preference-based systems, b-matching, acyclicity, round robin, adversarial and round robin daemons.

1 Introduction

A system is called *preference-based* if each of its nodes selfishly tries to activate its best edges according to some personal ranking. The description of the stable configurations — if any — of a p.b.s. is known as the stable b-matching problem. b-matching theory and its variants have applications in a variety of real-world situations, including dating agencies, college admissions, roommates attributions, assignment of graduating medical students to their first hospital appointment, or kidney exchanges programs [1,2,3,4,5].

Recently, Lebedev *et al.* showed that many preferences are acyclic. They also proved that an acyclic p.b.s. has a unique stable configuration, and always stabilizes[6,7]. This convergence result gave a theoretical proof for upload/download correlations in incentive-based networks like BitTorrent [8].

Our contribution is to analyze the convergence properties of acyclic p.b.s. by using the self-stabilization approach [9,10]. We use the convergence theorem proved in [6] to compute the step and round complexities for the round robin and the adversarial daemon.

The rest of this paper is organized as follows. In Section 2 we give a short introduction to the p.b.s. model and present the convergence theorem. In Section 3 we give the round complexity. The step complexity for the round robin and adversarial daemons are presented in Section 4.

2   Model

A preference-based system consists of a set V of n nodes, whose possible interactions are described by an acceptance graph G and a set of rankings $(<_i)_{i \in V}$.

The acceptance graph $G = (V, E)$ is an undirected graph. It describes compatibilities: a node i and a node j are capable of collaborating (we say that i is acceptable for j, and vice versa) if, and only if (iff) $\{i, j\} \in E$.

The ranking $<_i$ of a node i is a total order on the neighbors of i. If j and k are two distinct neighbors of i, then we say that i prefers j to k iff $j <_i k$.

Preference-based systems are also characterized by a quota vector b: a node i cannot sustain more that $b(i)$ simultaneous collaborations[1]. In this paper, we focus on the case $b = 1$, thus the state of collaborations at a given time is a matching of G, that we call configuration. $C(i)$ denotes the match of a node i, if any, in a configuration C.

The only action allowed in the model is the resolution of *blocking pairs*. A blocking pair is a pair of nodes that are not matched together, in which both could gain by collaborating (even if it means dropping the current match). A configuration with no blocking pair is *stable*.

We say that a node is *eligible* in a configuration if it belongs to at least one blocking pair. In this paper, we only consider *best-match* resolution: the action of an eligible node consists in choosing the node it prefers among the blocking pairs to which it belongs.

A *computation* is a maximal sequence of configurations such that for each configuration C_i, the next configuration C_{i+1} is obtained by the action of one eligible node in C_i. Maximality of a computation means that the computation is infinite or it terminates in a configuration where none of the actions are enabled, e.g. a stable configuration.

A *daemon* is an action scheduler that generates a computation. We consider here two kinds of daemons: the *round robin* daemon follows a round robin scheduling over V, and executes the action of i whenever i is selected by the scheduler and eligible; the adversarial daemon may select *any* eligible node at every step. The adversarial daemon encompasses the round robin.

2.1   Convergence Theorem for Acyclic p.b.s.

A preference cycle is a cycle of nodes $i_1, \dots, i_k$, with $k \geq 3$, such that each node i_j prefers its successor i_{j+1} to its predecessor i_{j-1} ($i_{j+1} <_{i_j} i_{j-1}$). A p.b.s. without any preference cycle is called *acyclic*.

A particular kind of acyclic p.b.s. are the global-preferences systems, where the preferences come from an inherent total order. For any global-preferences system, there exists a labeling $i_1, \dots, i_n$ of its nodes such that for any node i_j with distinct neighbors i_k and i_l, $i_k <_{i_j} i_l$ is equivalent to $k < l$. So, for sake of simplicity, the nodes of a global-preferences system are often labeled with the integers 1 to n.

---

[1] The b letter in *b-matching* stands for this quota vector.

In [6], Lebedev *et al.* proved that an acyclic p.b.s. has a unique stable configuration, and that all computations are finite. In other word, for any starting configuration, the adversarial daemon eventually converges to the stable configuration. Thus we can say that acyclic p.b.s. are self-stabilizing for their stable configuration.

Strong pairs. Strong pairs are a central concept in acyclic preference-based systems. A pair is strong in a configuration if:

- each node of the pair is ranked first by the other,
- or if there is one or more nodes that are ranked better, each of them is matched and forms a strong pair with its mate.

Strong pairs have many qualities (cf [6,7] for details):

- they are edges of the stable configuration,
- a strong blocking pair is stable until the pair is matched (the property is not affected by any daemon),
- the best-match resolution of any of the two nodes of a strong blocking pair matches these two nodes,
- once matched, a strong blocking pair is a stable matched pair,
- all non-stable configurations of a given acyclic p.b.s. admit at least one strong blocking pair.

Using strong blocking pairs, the question we will address in the following is the effective time complexity of the convergence under possibly adversarial scheduling regimes. We consider two measures for evaluating this complexity. The *round* complexity and the *step* complexity. A round is a sub-sequence of a computation in which every node that was eligible at the beginning of the round either is activated or ceases to be eligible during the round. The *step* complexity investigates the maximum length of a computation for all possible starting configurations.

3   Round complexity

The round complexity is simple to compute in the case of acyclic p.b.s., as shown by theorem 1:

Theorem 1. *Starting from any configuration, and under any daemon, an acyclic preference-based system stabilizes in $\lfloor \frac{n}{2} \rfloor$ rounds. Furthermore, there are instances where the round robin daemon requires $\lfloor \frac{n}{2} \rfloor$ rounds.*

Proof. The proof comes from the existence of strong blocking pairs in acyclic preference-based systems: after each round (until stabilization), we are sure that at least a a node of a strong blocking pair is selected, thus a stable edge is formed. As the stable configuration is a matching, it has at most $\lfloor \frac{n}{2} \rfloor$ edges, so the stabilization cannot last more than $\lfloor \frac{n}{2} \rfloor$ rounds. This bound is tight because it is reached for global preferences and complete acceptance graph, if we consider the round robin daemon with scheduling $n, (n-1), \ldots, 2, 1$, starting from the empty configuration $C_\emptyset$.

4   Step Complexity

In [6], the proof of the existence of a stable configuration relies on the following: in an acyclic p.b.s., all configurations of a given computation are distinct. Thus a first upper bound for the step complexity is the maximal number of matchings of a graph with n nodes. This number, also known as the number of involutions of a set of size n, has a factorial-like asymptotic behavior [11]. In the following we prove a tight quadratic bound for the round robin daemon and a tight exponential bound for the adversarial daemon.

4.1   Round Robin Daemon

The step complexity for the round robin daemon is given by theorem 2:

Theorem 2. *Starting from any configuration, the round robin daemon takes at most $\sum_{k=0}^{\lfloor \frac{n}{2} \rfloor - 1} (n - (2k + 1))$ steps to converge (hence the complexity is equivalent to $\frac{n^2}{4}$). This bound is tight.*

Proof. The reasoning is the same as for the round complexity. As long as the current configuration is not stable, there exists at least two nodes that belong to a strong blocking pair. Hence after at most $n - 1$ steps, the round robin daemon is forced to match a strong blocking pair. The remaining non stable nodes are less than $n - 2$, and if the configuration is not stable yet, at least two of them form a strong blocking pair, so after at most $n - 3$ steps, a new stable edge is formed...

If we continue this process, we see that the number of steps cannot be more than

$$(n - 1) + (n - 3) + (n - 5)... = \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor - 1} (n - (2k + 1)).$$

This is equal to $\frac{n^2}{4} + c$, where c is $-\frac{1}{4}$ when n is odd and 0 when n is even.

Like for the round complexity, one can see that this bound is reached for global preferences and complete acceptance graph, if we consider the round robin daemon with scheduling $n, (n - 1), \ldots, 2, 1$, starting from $C_\emptyset$.

4.2   Adversarial Daemon

The step complexity is harder to compute for the adversarial daemon. First we will prove that the step complexity is not greater than $2^{n-1} - 1$. Then we will introduce a daemon with complexity $\Omega(\mu^n)$, with $\mu \approx 1.6826$, thus proving that the complexity of the adversarial daemon stands somewhere between these two bounds.

2^n Upper Bound

Theorem 3. *The number of steps under any daemon is less than $2^{n-1} - 1$ for an acyclic preference-based system for all possible initial configurations.*

Proof. We need to introduce $S(n, k)$, that denotes the maximal number of steps that can be made by using only a fixed subset of size k of the nodes of an acyclic system of length n (the system and its initial configuration are arbitrary). $S(n, n)$ is the maximal complexity, so we want to prove that $S(n, n) \leq 2^{n-1} - 1$.

Let C be an initial configuration of a system of size n. If C is not stable, C admits at least one strong blocking pair. Thus any computation can be split in three subsequences: before, during, and after the resolution of the strong blocking pair. The two nodes of the strong blocking pair cannot be selected before or after the pair is resolved, so we have:

$$S(n, n) \leq \underbrace{S(n, n-2)}_{\text{before resolution}} + \underbrace{1}_{\text{resolution}} + \underbrace{S(n, n-2)}_{\text{after resolution}}$$

Now, we need to express $S(n, k)$ for $1 \leq k < n$. First, we have $S(n, 1) = 1$, because a single node i cannot be eligible after a selection: it is matched with the best node j it can (it cannot prefer the possible previous mate k of j, otherwise $\{i, j, k\}$ would be a preference cycle).

For $2 \leq k < n$, call A the set of nodes that can be selected, I the set of nodes that cannot be selected. One of the two following propositions is true:

- there exists a node in A that is never eligible as long as only nodes of A are selected,
- there exists a pair in $A \times A$ or in $A \times I$ that is strong with respect to the action of nodes from A.

In order to prove that, we consider the following path construction scheme: the successor, if any, of a node a in A is defined as the best choice for a that does not belong to a pair of I stable under the actions of A (stable nodes internal to I cannot interfere with other nodes, so one *forget* them). The successor of a non-stable node of I is its best choice among its neighbors from A plus its possible current neighbor. Note that a non stable node of I has always a successor.

Starting from an initial node a_0, one construct a path using this scheme. The successor of each node is the best node it can expect under the action of A. Because the system is acyclic and finite, the path eventually ends with a node without successor, or with two nodes j, k such that each one is the successor of the other.

If the path ends with a node without successor, this node belongs to A, and it cannot be eligible (its non-stable-neighbors list is empty). If it ends with two reciprocal successors i and j, then $\{i, j\}$ is strong under the action of A, and it belongs to $A \times A$ or $A \times I$ ($\{i, j\} \in I^2$ would imply that $\{i, j\}$ is a matched pair of I that is stable under the action of A).

We can now bound $S(n, k)$, using the same *before/during/after* argument than for $S(n, n)$:

$$S(n, k) \leq \begin{cases} S(n, k-1) & \text{if a node from } A \text{ is never eligible,} \\ 1 + 2S(n, k-2) & \text{if a pair from } A^2 \text{ is strong with respect to } A, \\ 1 + 2S(n, k-1) & \text{if a pair from } A \times I \text{ is strong with respect to } A. \end{cases}$$

In any case, we have $S(n, k) \leq 1 + 2S(n, k-1)$ for $2 \leq k < n$. An immediate recurrence gives $S(n, n-2) \leq 2^{n-2} - 1$, thus we have $S(n, n) \leq 2^{n-1} - 1$. This concludes the proof.

μ^n Lower Bound

Theorem 4. *There exists an acyclic preference-based system and a daemon such that the complexity is $\Omega(\mu^N)$, with*

$$\mu = \sqrt{\frac{\sqrt[3]{316 + 12\sqrt{249}}}{6} + \frac{20}{3\sqrt[3]{316 + 12\sqrt{249}}} + \frac{2}{3}} \approx 1.683$$

Proof. As the complexity is obviously increasing with n, we can restrain the proof to systems of even size. We consider a global-preferences system of size $N = 2n$ with complete acceptance graph.

The *worst eligible* daemon (w.e.d.) is defined, for global preference systems, as follows: as long as there exists an eligible node, the daemon select the worst-eligible node, i.e. the eligible node with the highest label.

Call T_C the number of steps taken by the w.e.d. to reach the stable configuration from an initial configuration C (the number of nodes and the knowledge graph are implicit). $T_\emptyset(n)$ stands for $T_{C_\emptyset(n)}$, where $C_\emptyset(n)$ is the empty configuration in the complete graph with $2n$ nodes. $T_{12}(n)$ stands for $T_{C_{12}(n)}$, where $C_{12}(n)$ is the stable configuration in the complete graph with $2n$ nodes, except the best pair, $\{1, 2\}$, is not matched.

We will prove that $T_\emptyset(n) = \Theta(\lambda^n)$, with $\lambda = \mu^2$. This will guarantee that the complexity of the adversarial worst-eligible daemon is $\Omega(\mu^N)$ (other initial configurations may take more steps).

$T_\emptyset$ can be expressed as a recursive function of T_{12}. This shown by Equation (1).

$$T_\emptyset(n) = n^2 + \sum_{i=1}^{n-1} (n-i)T_{12}(i) \tag{1}$$

For proving (1), we need to understand how the w.e.d. performs from $C_\emptyset$. The basic idea is that at each step of the w.e.d., nodes can be split according to a pivot. Nodes above the pivot or matched with those form the upper part. Other nodes form the lower part. Upper nodes are never selected by the daemon, while lower nodes perform the w.e.d. in a recursive way, with little change on upper part configuration. The pivot increases one by one until the stable configuration is reached.

We start with $C_\emptyset$ where all nodes are single.

$$\underbrace{\{2n\}, \{2n-1\}, \ldots, \{n+1\}}_{\text{lower part}}, \underbrace{\{n\}, \ldots, \{2\}, \{1\}}_{\text{upper part}}$$

First, $2n$ goes with 1. $2n$ is not eligible, so $2n-1$ is selected and takes 1 from $2n$, that goes to 2. Then $2n-2$ matches with 1, forcing $2n-1$ to lower its matching,

cascading to $2n$. After $\frac{n(n+1)}{2}$ steps, the w.e.d. reaches the configuration where each node of the lower part is matched with a node of the upper part as follows:

$$\{2n, \underbrace{n}_{\text{lowest eligible node}} \}, \{2n-1, n-1\}, \ldots, \{n+2, 2\}, \{n+1, 1\}$$

At this point, all nodes from what we have called the lower part are not eligible. So the first upper node, n, is selected and becomes the pivot, leading to the configuration

$$\underbrace{\{2n\}, \{n+1\},}_{\text{lower part}} \underbrace{\{2n-1, n-1\}, \ldots, \{n+2, 2\}, \{n, 1\}}_{\text{upper part}}$$

The lower part ($\{2n\}, \{n+1\}$) performs $T_\emptyset(1) = T_{12}(1)$ (in other words, $2n$ matches with $n+1$), then $n+1$ swaps with $n+2$ as 2's mate. The lower part is now $\{2n\}, \{n+2\}$ and performs $T_{12}(1)$, then $n+2$ swaps with $n+3$ as 3's mate, and so on... Eventually, $2n-2$ swaps with $2n-1$ as $n-1$'s mate and $2n$ matches with $2n-1$, resulting in the following configuration :

$$\{2n, 2n-1\}, \{2n-2, \underbrace{n-1}_{\text{lowest eligible node}} \}, \{2n-3, n-2\}, \ldots, \{n+1, 2\}, \{n, 1\}$$

Now $n-1$ is the lowest eligible node (l.e.n.) and becomes the pivot.

More generally, the configuration before $2 \leq i \leq n$ becomes the pivot is

$$\{2n, 2n-1\}, \ldots, \{2i+2, 2i+1\}, \{2i, \underbrace{i}_{l.e.n.} \}, \ldots, \{i+2, 2\}, \{i+1, 1\}$$

After i is selected, we obtain

$$\underbrace{\{2n, 2n-1\}, \ldots, \{2i+2, 2i+1\}, \{2i\}, \{i+1\},}_{\text{lower part}} \underbrace{\{2i-1, i-1\} \ldots, \{i+2, 2\}, \{i, 1\}}_{\text{upper part}}$$

The upper part needs will mutate $i-1$ times (including i's selection) to become

$$\{2i-2, i-1\} \ldots, \{i+1, 2\}, \{i, 1\}$$

Each mutation is due to the selection of the highest node from the lower part. This is what we call a transitional step (selection of a lower part node to match with an upper part node). After a transitional step, the lower part is made of $2(n-i+1)$ nodes; More precisely $2(n-i)$ nodes perform a local $\{2k, 2k-1\}$ matching, while the two best nodes (of the lower part) are single. As only the best node of the lower part can interact with the upper part and as the w.e.d. never selects the best node, the lower part will perform $T_{12}(n-i+1)$ steps before the next transitional step. To summarize, the w.e.d. computation, starting from $C_\emptyset$ is made of:

- $\frac{n(n+1)}{2}$ steps to match each node of the upper half to a node of the lower half.
- for each pivot i ($n \leq i \leq 2$), $i-1$ transitional steps, and $(i-1)T_{12}(n-i+1)$ low steps.

That leads to Equation (1) :

$$
\begin{aligned}
T_\emptyset(n) &= \frac{n(n+1)}{2} + \sum_{i=2}^{n}(i-1)(1 + T_{12}(n-i+1)) \\
&= \frac{n(n+1)}{2} + \frac{n(n-1)}{2} + \sum_{i=1}^{n-1}(n-i)T_{12}(i) \\
&= n^2 + \sum_{i=1}^{n-1}(n-i)T_{12}(i)
\end{aligned}
$$

Now that we have an expression of $T_\emptyset$ that depends on T_{12}, we need to specify the behavior of T_{12}. This is given by Equation (2).

$$
T_{12}(n) = \begin{cases} 0 & \text{if } n < 1 \\ 1 & \text{if } n = 1 \\ 4\left(n-1+\sum_{i=1}^{n-3}T_{12}(i)\right) + 3T_{12}(n-2) + T_{12}(n-1) & \text{if } n > 1 \end{cases} \quad (2)
$$

For $n \leq 1$, (2) is obvious. For $n \geq 2$, we adapt (1)'s proof. The w.e.d. first matches all upper nodes, then combinatorics can be made using transitional and low steps derived from a pivot. The main difference is that due to pre-existing upper matchings, we need to distinguish odd and even pivots. Now, let us describe the different phases of the w.e.d. computation starting from C_{12} :

Initial phase. First $2n$ matches 1, $2n-1$ matches 1 and $2n$ matches 2. So after 3 steps, nodes form the configuration

$$\{2n-2, 2n-3\}, \dots, \{4, 3\}, \{2n, 2\}, \{2n-1, 1\}$$

Our first pivot will be $2n-2$ (even).

Even pivot. The configuration before $2i$ ($n-1 \leq i \leq 2$)[2] becomes the pivot is:

$$\{2n, 2n-1\}, \dots, \{2i+4, 2i+3\}, \{2i, 2i-1\}, \dots, \{4, 3\}, \{2i+2, 2\}, \{2i+1, 1\}$$

After $2i$ is selected, we obtain

$$\underbrace{\{2n, 2n-1\}, \dots, \{2i+4, 2i+3\}, \{2i+1\}, \{2i-1\}}_{\text{lower part}}, \dots, \{2i+2, 2\}, \{2i, 1\}$$

The lower part is made of $2(n-i)$ nodes that form a C_{12} configuration (the two best nodes, $2i+1$ and $2i-1$, are single). Low transitions occurs until the second best node ($2i+1$) is selected and matches with 2. In other words, after $T_{12}(n-i) - T_{12}(n-i-1)$ transitions (we count the $T_{12}(n-i)$ transitions

---

[2] We treat the pivot 2 separately.

except we remove transitions that happens after node 2 is selected) we obtain the configuration

$$\underbrace{\{2n, 2n-1\}, \ldots, \{2i+4\}, \{2i+2\}, \{2i+3, 2i-1\}}_{\text{lower part}}, \ldots, \{2i+1, 2\}, \{2i, 1\}$$

The lower part configuration is similar to $C_{12}(n-i)$ except for the four best nodes[3] : first and third nodes ($2i-1$ and $2i+3$) are together, while second and fourth nodes are single. The w.e.d. first performs the stable solution for the lower part minus $2i-1$ and $2i+3$ in $T_{12}(n-i-1)$ steps:

$$\underbrace{\{2n, 2n-1\}, \ldots, \{2i+4, 2i+2\}, \{2i+3, 2i-1\}}_{\text{lower part}}, \ldots, \{2i+1, 2\}, \{2i, 1\}$$

Then $2i+2$ is selected and matches $2i-1$:

$$\underbrace{\{2n, 2n-1\}, \ldots, \{2i+4\}, \{2i+3\}, \{2i+2, 2i-1\}}_{\text{lower part}}, \ldots, \{2i+1, 2\}, \{2i, 1\}$$

Lastly, after $T_{12}(n-i-1)$ steps, the lower part is stabilized and we have

$$\{2n, 2n-1\}, \ldots, \{2i+4, 2i+3\}, \{2i+2, \underbrace{2i-1}_{\text{l.e.n.}}\}, \ldots, \{2i+1, 2\}, \{2i, 1\} \quad (3)$$

$2i-1$ is now the lowest eligible node and is ready to become the pivot.

Odd pivot. (3) is the configuration before $2i-1$ ($n-1 \le i \le 2$) becomes the pivot. After that step, the configuration is

$$\underbrace{\{2n, 2n-1\}, \ldots, \{2i+4, 2i+3\}, \{2i+2\}, \{2i\}}_{\text{lower part}}, \ldots, \{2i+1, 2\}, \{2i-1, 1\}$$

After $T_{12}(n-i)$ steps, the w.e.d. stabilizes the lower part:

$$\underbrace{\{2n, 2n-1\}, \ldots, \{2i+4, 2i+3\}, \{2i+2, 2i\}}_{\text{lower part}}, \ldots, \{2i+1, 2\}, \{2i-1, 1\}$$

Then $2i$ swaps with $2i+1$ as 2's mate:

$$\underbrace{\{2n, 2n-1\}, \ldots, \{2i+4, 2i+3\}, \{2i+2\}, \{2i+1\}}_{\text{lower part}}, \ldots, \{2i, 2\}, \{2i-1, 1\},$$

---

[3] For $i = n-1$, there is only two nodes involved. However, the presented results stand if we set $T_{12}(n) = 0$ for any $n \le 0$.

and after another $T_{12}(n-i)$ steps, the configuration is

$$\{2n, 2n\text{–}1\}, \ldots, \{2i\text{+}4, 2i\text{+}3\}, \{2i\text{+}2, 2i\text{+}1\}, \underbrace{\{2i\text{–}2}_{\text{l.e.n.}}, 2i\text{–}3\}, \ldots, \{2i, 2\}, \{2i\text{–}1, 1\}.$$

This is the point where $2i-2$ becomes the next pivot

2 as a pivot. Eventually, 2 is selected and matches with 1. After that, the w.e.d. performs $T_{12}(n-1)$ to produce the stable configuration.

We can now verify Equation (2):

$$T_{12}(n) = \underbrace{3}_{\text{Initial phase}} + \underbrace{\sum_{i=2}^{n-1} (2 + T_{12}(n-i) + T_{12}(n-i-1))}_{\text{Even pivots}}$$

$$+ \underbrace{\sum_{i=2}^{n-1} (2 + 2T_{12}(n-i))}_{\text{Odd pivots}} + \underbrace{1 + T_{12}(n-1)}_{\text{2 as a pivot}}$$

$$= 4 + T_{12}(n-1) + \sum_{i=2}^{n-1} (4 + 3T_{12}(n-i)) + \sum_{i=3}^{n} (T_{12}(n-i))$$

$$= 4(n-1) + \sum_{i=1}^{n-3} 4T_{12}(i) + 3T_{12}(n-2) + T_{12}(n-1)$$

Equation and (2) allows us to affirm that T_{12} follows an asymptotic geometric progression. Moreover, the asymptotic common ratio can be explicitly found: it must be a positive solution of Equation (4).

$$x^2 = x^1 + 3 + \frac{4}{x-1} \tag{4}$$

This equation leads to Equation (5), a third degree equation whose only positive root is the asymptotic common ratio for T_{12}.

$$x^3 = 2x^2 + 2x + 1 \tag{5}$$

Analysis indicates that the positive root of Equation (5) is

$$\lambda = \frac{\sqrt[3]{316 + 12\sqrt{249}}}{6} + \frac{20}{3\sqrt[3]{316 + 12\sqrt{249}}} + \frac{2}{3} \approx 2.8312$$

Using Equation (1), we see that $T_{\emptyset}$ is also $\Theta(\lambda^n)$. That ends our proof.

5   Conclusion

We have shown that acyclic preference-based systems have a linear round complexity even for the adversarial daemon, and a quadratic step complexity for the round robin daemon. This means the the self-stabilization of such systems is good, as long as nodes cannot be eligible and not selected for an arbitrary

long period of time. These bounds are tight for global p.b.s., but according to a previous work, the round complexity may be logarithmic for most acyclic p.b.s. that are not global [12]. On the other hand, the step complexity stands between $\Omega(\mu^n)$ and $2^{n-1} - 1$ for the adversarial daemon. This is a more precise result than the factorial-like upper bound that can be deduced from the convergence theorem. Note, that global p.b.s. with complete acceptance graph have been used whenever we needed to prove that a bound was reached. Thus the global p.b.s. with complete acceptance graph is a sort of extremum among the possible acyclic p.b.s., as empirically observed in [12].

References

1. Gale, D., Shapley, L.: College admissions and the stability of marriage. American Math. Monthly 69, 9–15 (1962)
2. Irving, R.W., Manlove, D., Scott, S.: The hospitals/residents problem with ties. In: Halldórsson, M.M. (ed.) SWAT 2000. LNCS, vol. 1851, pp. 259–271. Springer, Heidelberg (2000)
3. Irving, R.W., Manlove, D.F.: The stable roommates problem with ties. J. Algorithms 43(1), 85–105 (2002)
4. Roth, A.E.: The evolution of the labor market for medical interns and residents: A case study in game theory. Journal of Political Economy 92(6), 991–1016 (1984)
5. Roth, A.E., Sonmez, T., Utku Unver, M.: Pairwise kidney exchange. Journal of Economic Theory 125(2), 151–188 (2005)
6. Lebedev, D., Mathieu, F., Viennot, L., Gai, A.T., Reynier, J., Montgolfier, F.D.: On using matching theory to understand p2p network design. In: INOC (2007)
7. Gai, A.T., Lebedev, D., Mathieu, F., de Montgolfier, F., Reynier, J., Viennot, L.: Acyclic preference systems in p2p networks. In: Euro-Par (2007)
8. Gai, A.T., Mathieu, F., Reynier, J., De Montgolfier, F.: Stratification in P2P networks application to bittorrent. In: ICDCS (2007)
9. Dijkstra, E.W.: Self-stabilizing systems in spite of distributed control. Commun. ACM 17(11), 643–644 (1974)
10. Dolev, S.: Self-Stabilization. MIT Press, Cambridge (2000)
11. Chowla, S.: The asymptotic behavior of solutions of difference equations. In: Proceedings of the International Congress of Mathematicians, vol. I, 377, Amer. Math. Soc (1950)
12. Mathieu, F.: Self-stabilization in preference-based networks. In: P2P (2007)