

# Open Problems

GRASTA-MAC 2015

## 1 Exploration with Stop in Ring with limited visibility (Franck Petit)

Consider autonomous, identical, oblivious robots, in the *Look-Compute-Move* model. They have only local visibility, that is the snapshot allows them to see only at distance  $\Phi$ . In particular, they don't know the size of the ring nor the number of robots if  $\Phi < n/2$ .

The main question is: what is the minimum number of robots required to explore the ring with termination (in a deterministic way)?

**Known results:**

- $\Phi = 1$ : impossible in semi-synchronous setting. Possible in a synchronous setting with 5 robots.
- $\Phi = 2$ : algorithm for 7 asynchronous robots starting from a strongly connected configuration (robots occupying consecutive nodes).
- $\Phi = 3$ : algorithm for 5 asynchronous robots from a strongly connected configuration (it is optimal). algorithm for 7 asynchronous robots from a weak connected configuration (the visibility graph must be connected).

**Questions:**

- What is the minimum number for  $\Phi = 2$  from a strongly connected configuration?
- Algorithm starting from a weakly connected configuration?
- Is  $\Phi = 3$  as strong as  $\Phi = \infty$ ?
- Algorithm for general number of robots?
- ...

## 2 Moderate exponential time algorithms (Pierre Fraigniaud)

Most of the problems considered in the workshop (graph searching, cops-and-robber, exploration, etc.) yield optimization problems that are at least NP-hard. This prevents us from designing polynomial-time algorithms for solving these problems (unless unexpected complexity results holds). However, this does not

prevent us from designing algorithms that are still efficient for relatively small instances (few number of agents, and/or small graphs).

The open problem is the design of moderately exponential algorithms for some of the problems considered in the workshop (i.e., algorithms with complexity of the form, e.g.,  $(1 + \epsilon)^n$  or  $2^{o(n)}$ ). Such algorithms could have an interest beyond our community, as the problems considered in the workshop are of a “dynamic” nature that is very different from the classical “static” combinatorial problems. Moreover, having algorithms at hand allowing us to solve small instances could help providing intuition about how to solve larger instances, and/or to elaborate conjectures about the solutions of large instances.

### 3 Synchronization (Christian Scheideler)

The question is about the formal analysis of the following gathering algorithms.

**On a ring:** Consider  $n$  robots on a cycle. At each step, each robot chooses two robots uniformly at random (including itself) and then moves to the position of the closer robot one. (positions of the robots correspond to their clock, and the goal is to synchronize them, i.e., when a robot moves toward a robot, it adjusts its clock offset to the offset of the other robot).

Simulations show that the process converge in time  $O(\log n)$  but no formal analysis.

**On the plane:** Consider  $n$  robots on the plane. At each step, each robot chooses 2 robots uniformly at random (including itself) and goes to the closest one. Simulations show that the process converge in time  $O(\log n)$  but no formal analysis.

There is exists a formal analysis of such a process on the line, if the robots chooses 3 robots uniformly at random (including itself) and goes to the middle one.

**Reference:** [Benjamin Doerr, Leslie Ann Goldberg, Lorenz Minder, Thomas Sauerwald, Christian Scheideler: Stabilizing consensus with the power of two choices. SPAA 2011: 149-158]

### 4 Flag game (Ben Seamone)

Let  $G = (V, E)$  be a graph,  $v_0 \in V$ .  $F \subset V$  be the set of *flags*. Let us consider the following two-player game. The rule are similar to the ones of Cops and Robber game, but now the Robber starts from  $v_0$  and wins if it reaches a node in  $F$  before being caught by the cops.

The questions are about combinatorial bounds and algorithmic complexity on the minimum number number of cops needed to win.

### 5 About Permutations (Arnaud Labourel, from Peter Widmayer)

What is the complexity of the following problem?

Given a non-decreasing sequence  $\{v_1, \dots, v_m\}$  of  $m$  integers, is there a permutation  $\pi$  of  $\{1, \dots, m+1\}$  such that, for any  $1 \leq i \leq m$ ,  $\pi(i) + \pi(i+1) \leq v_i$ ?

If the sequence is arbitrary, the problem is known to be NP-complete.

## 6 Rugby (Nicolas Nisse)

Consider an  $n \times \infty$  grid. We consider the following turn-by-turn game with two players: the attacker has speed two, i.e., it may move along two edges at each turn, the defender controls  $k$  agents with speed one.

What is the minimum number  $k$  of defenders necessary to avoid that the attacker starting on the “left” side of the infinite grid manage to reach the “right side” of it without being caught (he is blocked if one defender occupies the same vertex as it)?

Right now, it is known that  $\log(n) \leq k \leq n/6$  [M. Mamino].

## 7 Searching in Boxes (Thomas Lidbetter)

Consider a set  $N = \{1, \dots, n\}$  of  $n$  boxes. For any  $i \leq n$ , let  $c_i$  be the cost of searching a box (opening it). The boxes may contain some balls and the goal is to minimize the expected cost of finding all balls.

**Case of one ball.** Let  $p_i$  be the probability that box  $i$  contains the ball. Note that  $\sum_{i \leq n} p_i = 1$  since there is a unique ball. In that case, the optimal solution is obtained by the following greedy algorithm: search the boxes in the non increasing ordering of  $c_i/p_i$ . (Smith’s rule, 1956).

**Case of two balls.** Let  $p_{i,j}$  be the probability that the balls are in boxes  $i$  and  $j$ . The following greedy algorithm provides a 2-approximation: start with the set  $A \subseteq N$  which minimize  $c(A)/p(A)$  (such a set  $A$  can be computed in polynomial-time by submodularity).

Can we do better?

## 8 Fast Capture (Gena Hahn)

Consider the original (Nowakowski-Winkler, Quilliot) cops and robber game. For a graph  $G$  (assumed connected), write  $cn(G)$  for the cop number of  $G$ , the minimum number of cops that always catch the robber. Let  $ct(G)$  be the capture time of  $G$ , that is, the minimum number of rounds the  $cn(G)$  cops need to capture the robber in the worst case.

**Problem 1.** *Is there a better method to find, or even estimate,  $ct(G)$  for a given graph  $G$  known than using the Hahn-MacGillivray algorithm to determine  $cn(G)$  and reading the capture time off the labeling of the vertices of the auxilliary graph (which can be of order exponential in the number of vertices of  $G$ )?*

**Problem 2.** *To be more realistic, one could ask that the robber be caught within a certain time  $t$ . Define  $ct_t(G)$  to be the minimum number of cops needed to capture the robber in at most  $t$  rounds in the worst case. What can we say about it (beside the obvious  $ct_0(G) = n$  and  $ct_1(G) = dn(G)$ , with  $dn(G)$  being the size of a minimum dominating set of  $G$ )?*

## 9 Burning Number (N. Nisse)

Let  $G = (V, E)$  be a graph, we are interested in the minimum  $k$  such that there exists a sequence  $(v_1, \dots, v_k)$  of vertices with  $V = \bigcup_{i \leq k} N_{k-i}(v_i)$ , where  $N_r(v)$  denotes the set of vertices at distance at most  $r$  from  $v$ . Such a minimum  $k$  is called the *burning number* of  $G$  and denoted by  $bn(G)$ .

It is known that computing  $bn(G)$  is strongly NP-complete and that there exists a 3-approximation for it. Can we do better? (F)PTAS ?

It is known that  $bn(G) \leq \lceil \sqrt{\frac{3}{2}n} \rceil$  for any connected graph  $G$ .

**Conjecture:**  $bn(G) \leq \lceil \sqrt{n} \rceil$  for any connected graph  $G$ . (Note that it is sufficient to prove it for trees)