Agreement with oblivious robots

Maria Gradinariu Potop-Butucaru
Université Pierre et Marie Curie
LIP6, Inria, France

The agreement invariant

- to share the same (an approximate) location
  - gathering (convergence)
- to not share the same location
  - scattering
- to acknowledge the same cheef
  - leader election
- to form and maintain the same pattern
  - flocking, pattern formation, connectivity

Agreement QoS

- Agreement time
- Faults resilience
- Scheduler freedom

Agreement feasibility

- Without additional assumptions exact agreement is impossible to achieve

Tools to bypass the impossibility results

- increased knowledge (multiplicity, common axes/origin/orientation)
- constrained scheduler (bounded, fair)
- constrained number of faulty robots
- random choices

The GATHERING case

- Acknowledgements
  - Ando, Oasa, Suzuki, Yamashita [Trans. on Robotics and Automation 1999]
  - Flocchini, Prencipe, Santoro, Widmayer [TCS 2005]
  - Agmon and Peleg [SIAM J. of Computing 2006]
  - Cohen and Peleg [STACS 2006]
  - Dieudonné and Petit [FUN2007]
  - Defago, Gradinariu, Messika, Raipan-Parvédy [DISC 2006 + new results]
  - Clémant, Defago, Gradinariu, Izumi [IPL 2010]

Fault-free environement

Scheduler	multiplicity			
	deterministic		probabilistic	
	$n=2$	$n \geq 3$	$n=2$	$n \geq 3$
unfair	NO(*L.1*)	NO(*L.10*)	**OK(L.8)**	**OK(L.18)**
unfair centr.	**OK(L.9)**	NO(*L.10*)	OK(*L.8*)	**OK(L.18)**
fair	**NO(L.1)**	**OK(L.2)**	OK(*L.8*)	OK(*L.2*)
fair centr.	OK(*L.9*)	OK(*L.2*)	OK(*L.8*)	OK(*L.2*)
fair bounded		OK(*L.2*)	OK(*L.8*)	OK(*L.2*)
fair *k*-bounded		OK(*L.2*)	OK(*L.8*)	OK(*L.2*)
fair bounded reg.		OK(*L.2*)	OK(*L.8*)	OK(*L.2*)
fair centr. bounded reg.	OK(*L.9*)	OK(*L.2*)	OK(*L.8*)	OK(*L.2*)

Scheduler	without multiplicity			
	deterministic		probabilistic	
	$n=2$	$n \geq 3$	$n=2$	$n \geq 3$
unfair	NO(*L.3*)	NO(*L.3*)	**OK(L.8)**	NO(*L.12*)
unfair centr.	**OK(L.9)**	NO(*L.7*)	OK(*L.8*)	NO(*L.12*)
fair	**NO(L.3)**	**NO(L.3)**	OK(*L.8*)	NO(*L.12*)
fair centr.	OK(*L.9*)	NO(*L.7*)	OK(*L.8*)	**NO(L.12)**
fair bounded		NO(*L.7*)	OK(*L.8*)	**OK(L.13)**
fair *k*-bounded		NO(*L.7*)	OK(*L.8*)	OK(*L.13*)
fair bounded reg.		NO(*L.7*)	OK(*L.8*)	OK(*L.13*)
fair centr. bounded reg.	OK(*L.9*)	**NO(L.7)**	OK(*L.8*)	OK(*L.13*)

Probabilistic gathering

Functions:
observe_neighbors :: returns the set of robots within visibility range of robot p (the set of p's neighbors). Note that, in a system with unlimited visibility, *observe_neighbors* returns all the robots in the network.

Actions:

$\mathcal{A}_1$ :: *true* $\longrightarrow$

  $\mathcal{N}_p =$ *observe_neighbors*();
  with probability $\alpha = \frac{1}{|\mathcal{N}_p \bigcup \{p\}|}$ do
    select a robot $q \in \mathcal{N}_p \bigcup \{p\}$;
    move towards q;
  Remark: with probability $1 - \alpha$, *the position remains unchanged*;

- convergence in $O(n^2)$ rounds when strong multiplicity knowledge is used
- no multiplicity knowledge: exponential convergence time

Probabilistic gathering

Functions: *observe_neighbors* :: returns the set of robots in the system
maximal_multiplicity :: returns the set of points with maximal multiplicity
local_position :: returns the local position of the robot p
(i.e., the locations shared by the largest number of robots as returned by the observe function);

Actions:
$\mathcal{N}_p = observe\_neighbors()$;
if $|maximal\_multiplicity(\mathcal{N}_p)| > 1$ **then**
  with probability $\frac{1}{2n}$
    let q be a point chosen arbitrarily from $maximal\_multiplicity(\mathcal{N}_p) \setminus lp()$;
    move towards q;
  with probability $1 - \frac{1}{2n}$
    do not move;
if $|maximal\_multiplicity(\mathcal{N}_p)| = 1$ **then**
  let q be the point given by $maximal\_multiplicity(\mathcal{N}_p)$;
  move towards q;

- convergence in O(1) rounds
- strong multiplicity knowledge

Probabilistic gathering

Actions:

$\mathcal{N}_p = observe\_neighbors()$;
if $\mathcal{N}_p$ includes two or more multiplicity points **then**
    Execute Algorithm 6.1;
else if $\mathcal{N}_p$ includes a single multiple point q **then**
    Move towards q;
else
    let q be a point chosen arbitrarily from $\mathcal{N}_p$;
    With probability $\frac{1}{2n}$
        move towards q;
    With probability $1 - \frac{1}{2n}$
        do not move;

- convergence in O(log n log(log n)) rounds
- weak multiplicity knowledge
- use a simple scattering scheme

Two scattering schemes

Weak multiplicity knowledge O(log n loglog(n))

Compute the Voronoi diagram;
$Cell_i$:= Voronoi cell where r_i is located;
$Current\_Pos$:= position where r_i is located;

if $Random() = 0$ **then**
Move toward an arbitrary position in $Cell_i$, different from $Current\_Pos$;
else Do not move;

Strong multiplicity knowledge O(1)

Compute the Voronoi diagram;
$Cell_i$:= Voronoi cell where r_i is located;
$Current\_Pos$:= position where r_i is located;

Let Pos be a set of $2n^2$ positions in $Cell_i$:
Move toward a position in Pos chosen uniformly at random,

Two lessons

- refining the multiplicity knowledge may improve the complexity
- refining the scheduler may change de history

Crash-prone environement

Scheduler	multiplicity		without multiplicity	
	deterministic	probabilistic	deterministic	probabilistic
		$f = 1$		
unfair		**OK(L.18)**	NO(*L.3*)	NO(*L.12*)
unfair centr.		OK(*L.18*)	NO(*L.7*)	NO(*L.12*)
fair	!!!**OK(L.4)**!!!	OK(*L.4*)	NO(*L.3*)	NO(*L.12*)
fair centr.	OK(*L.4*)	OK(*L.4*)	NO(*L.7*)	NO(*L.12*)
fair bounded	OK(*L.4*)	OK(*L.4*)	NO(*L.7*)	OK(*L.15*)
fair *k*-bounded	OK(*L.4*)	OK(*L.4*)	NO(*L.7*)	OK(*L.15*)
fair bounded reg.	OK(*L.4*)	OK(*L.4*)	NO(*L.7*)	OK(*L.15*)
fair centr. bounded reg.	OK(*L.4*)	OK(*L.4*)	NO(*L.7*)	OK(*L.15*)
		$f \geq 2$		
unfair		**OK(L.18)**	NO(*L.16*)	NO(*L.16*)
unfair centr.		OK(*L.18*)	NO(*L.16*)	NO(*L.16*)
fair		OK(*L.18*)	NO(*L.16*)	NO(*L.16*)
fair centr.	**OK(L.17)**	OK(*L.17*)	NO(*L.16*)	NO(*L.16*)
fair bounded		OK(*L.18*)	NO(*L.16*)	NO(*L.16*)
fair *k*-bounded		OK(*L.18*)	NO(*L.16*)	NO(*L.16*)
fair bounded reg.		OK(*L.18*)	NO(*L.16*)	NO(*L.16*)
fair centr. bounded reg.	OK(*L.17*)	OK(*L.17*)	**NO(L.16)**	**NO(L.16)**

Byzantine-prone environement

Scheduler	multiplicity; deterministic					
	$f=1$			$f \geq 2$		
	$n=3$	$n \geq 4$ (even)	$n \geq 4$ (odd)	$n=3$	$n \geq 4$ (even)	$n \geq 4$ (odd)
unfair	NO(*L.5*)	NO(*L.5*)	NO(*L.5*)	NO(*L.5*)	NO(*L.5*)	NO(*L.5*)
unfair centr.		NO(*L.21*)		NO(*L.22*)	NO(*L.21*)	NO(*L.22*)
fair	**NO(L.5)**	NO(*L.5*)	NO(*L.5*)	NO(*L.5*)	NO(*L.5*)	NO(*L.5*)
fair centr.		NO(*L.21*)	**OK(N.2)**	NO(*L.22*)	NO(*L.21*)	NO(*L.22*)
fair bounded		NO(*L.21*)		NO(*L.22*)	NO(*L.21*)	NO(*L.22*)
$(k \geq n-1)$-bounded		NO(*L.21*)		NO(*L.22*)	NO(*L.21*)	NO(*L.22*)
$(\Gamma(n,f) \leq k \leq n-2)$-bounded		**OK(C.4)**		NO(*L.22*)	NO(*L.21*)	NO(*L.22*)
$(2 \leq k < \Gamma(n,f))$-bounded		**OK(C.4)**				
fair bounded reg.		OK(*C.4*)				
centr. $(k \geq n-1)$-bound.		**NO(L.21)**	OK(*N.2*)	NO(*L.22*)	NO(*L.21*)	NO(*L.22*)
centr. $(\Gamma(n,f) \leq k \leq n-2)$-bound.		OK(*C.4*)	OK(*N.2*)	**NO(L.22)**	**NO(L.21)**	**NO(L.22)**
centr. $(2 \leq k < \Gamma(n,f))$-bound.		OK(*C.4*)	OK(*N.2*)			
fair centr. bounded reg.		OK(*C.4*)	OK(*N.2*)			
fully synchronized		**OK(L.6)**	**OK(L.6)**		**OK(L.6)** if $n \geq 3f+1$	

$\Gamma(n,f) = \left\lceil \frac{n-f}{f} \right\rceil$ if n even ; $\Gamma(n,f) = \left\lceil \frac{n-f}{f-1} \right\rceil$ if n odd.

Byzantine-prone environement

Functions:
observe_neighbors :: returns the set of robots within the vision range of robot p (the set of p's neighbors);
maximal_multiplicity :: returns the set of robots with the maximal multiplicity;

Actions:

$\mathcal{A}_1$ :: *true* $\longrightarrow$
    $\mathcal{N}_p = observe\_neighbors()$;
    if $p \in maximal\_multiplicity(\mathcal{N}_p) \wedge |maximal\_multiplicity\ (\mathcal{N}_p)| > 1$ then
        with probability $\frac{1}{|maximal\_multiplicity(\mathcal{N}_p)|}$ do
            select a robot $q \in maximal\_multiplicity(\mathcal{N}_p)$;
            move towards q;
    else
        select a robot $q \in maximal\_multiplicity(\mathcal{N}_p)$;
        move towards q;

- strong multiplicity knowledge
- probabilistic scheduler
- exponential convergence time !!!!!

Byzantine resilient CONVERGENCE

Computation Model	Scheduler	Bounds
ATOM	fully synchronous	$n > 3f$
ATOM	fully synchronous	$n > 2f$
ATOM	k-bounded	$n > 3f$
CORDA	k-bounded	$n > 4f$
CORDA	k-bounded	$n > 3f$
CORDA	**fully asynchronous**	$n > 5f$

- Acknowledgements
  - Cohen and Peleg
  - Zohir Bouzid and Sébastien Tixeuil

Open problems

- Byzantine resilient gathering
  - investigate the power of probabilistic algorithms
- Byzantine resilient convergence
  - multi-dimensional spaces
- connections with classical distributed computing area
  - convergence ~ aproximate consensus
  - scattering ??? renaming
  - gathering ??? consensus / k-set agreement
  - robot models ??? distributed models