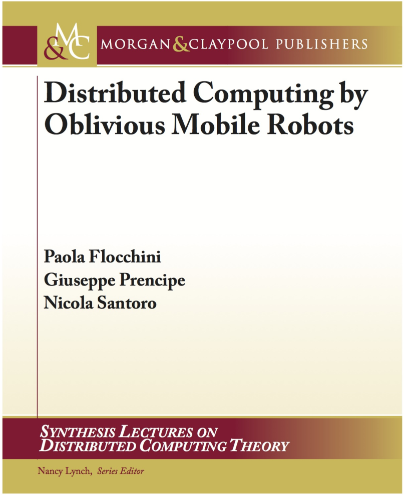
Theory and practice of distributed robotics

Uli Fahrenberg

LRE & EPITA Rennes

Submeeting 2023

A nice book


MORGAN & CLAYPOOL PUBLISHERS
Distributed Computing by Oblivious Mobile Robots
Paola Flocchini
Giuseppe Prencipe
Nicola Santoro
SYNTHESIS LECTURES ON DISTRIBUTED COMPUTING THEORY
Nancy Lynch, Series Editor

Summary

- Lots of research in the theory of distributed robotics.
- Very much related to the theory of distributed systems.
- Focus on autonomous simple robots with limited communication capabilities.
- Nice theoretical results, but few things have been done in practice.
- Practical distributed robotics seems to be very far away from this.

1 Introduction

2 Models

3 Results

4 Distributed Underwater Robotics?

5 Conclusion

Robot models

- Networks of mobile robots which are
  - anonymous: they all run the same algorithm
  - oblivious: they keep no memory of prior computations
  - distributed: there is no central control
  - with implicit communication (typically through light)
- Sometimes, "oblivious" is relaxed to "finite memory".
- Generally, freedom from failures is assumed; few works on robots with crash faults or Byzantine faults
- Generally, robots are assumed to
  - be dimensionless: points in space; few works on solid or fat robots
  - have infinite precision; few works on inaccurate robots
  - have no notion of real time

Robots have layers (like onions or cakes)

Two-layer control model:

- Layer 1: control of individual robots
- Layer 2: control of the network

(For Layer 1, another nice book.)

Veni vidi vici

- The Look-Compute-Move cycle:
  1. Look around and gather positions of other robots and obstacles
     - sometimes, limited visibility is assumed
  2. Compute your next move
     - with or without knowledge of previous positions or moves
  3. Move to the computed new position
     - or stay put if you wish
- No looking or computing during the Move phase!
- No real-time model: can't say how long the phases will be

Network models

- fully synchronous (FSYNC): all LCM cycles in lockstep
- semi-synchronous (SSYNC): all LCM cycles in lockstep, but in every round only a subset of robots participates
- asynchronous (ASYNC): most interesting (and difficult!)

Theorem

ASYNC $\subsetneq$ *SSYNC* $\subsetneq$ *FSYNC*

Theorem

ASYNC + *5-colored lights* $\supsetneq$ *SSYNC*

Theorem

ASYNC + *3-colored lights* + *one-snapshot memory* $\supsetneq$ *FSYNC*

Gathering and convergence

- Convergence: make robots meet in one point.
- Gathering: make robots meet in one point *in a finite number of rounds*.

Theorem

Gathering is solvable in FSYNC, even with restricted mobility.
Convergence is solvable in ASYNC, even with restricted mobility.

Proof.

Move to center of gravity. □

Theorem

Gathering is impossible in SSYNC (and hence in ASYNC).

Proof.

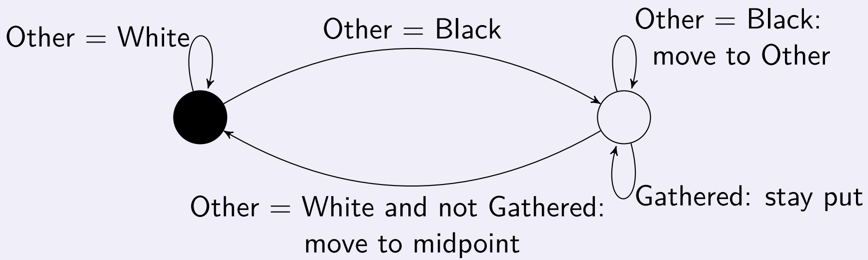
Move-to-CoG does not work; neither does anything else. □

Gathering with lights

Theorem (Heriban (COMASIC!), Défago, Tixeuil 2018)

Gathering 2 robots is solvable in ASYNC with 2-colored lights.

Proof.



□

Gathering solid robots

- Fail-stop collisions: if a robot collides with another during Move, it stops.
- Gathering: make robots form a connected configuration (in a finite number of rounds).

Theorem

Gathering is solvable in ASYNC for 2, 3 or 4 solid robots, in $\mathbb{R}^2$, assuming common unit distance and fail-stop collisions.

Proof.

(It's complicated.) □

(Nothing more seems to be known.)

Convergence with limited visibility

- Same visibility range for all robots.
- Visibility graph: points = robots; edge iff visible
- partial ASYNC: global time bound on LCM cycle duration

Theorem

Convergence is impossible in FSYNC if the initial visibility graph is disconnected.

Proof.

Trivial. □

Theorem

Convergence is solvable in partial ASYNC (and hence in SSYNC).

Proof.

Move towards center of circle which encloses all visible companions. □

Convergence with inaccuracies

- Distance imprecision ϵ: measurement $\subseteq [1-\epsilon, 1+\epsilon] \cdot$ distance
- Angular imprecision θ: $|\text{measurement} - \text{angle}| \leq \theta$

Theorem

Gathering is impossible in FSYNC with distance imprecisions, even with memory and randomness.

Proof.

Partition the line into finitely many segments of length $\frac{1+\epsilon}{1-\epsilon}$ . . . □

Theorem (Cohen-Peleg 2008)

Convergence is impossible in FSYNC if $\theta \geq 60^\circ$, even with unlimited memory.

Convergence with inaccuracies, contd.

- Distance imprecision ϵ: measurement $\subseteq [1-\epsilon, 1+\epsilon] \cdot$ distance
- Angular imprecision θ: $|\text{measurement} - \text{angle}| \leq \theta$

Theorem (Cohen-Peleg 2008)

Convergence is solvable in FSYNC if $\sqrt{2(1+\epsilon)(1-\cos\theta)+\epsilon^2} < 0.2$.

Proof.

Move to center of gravity, but stay outside circle of possible error. □

Conjecture (Cohen-Peleg 2008)

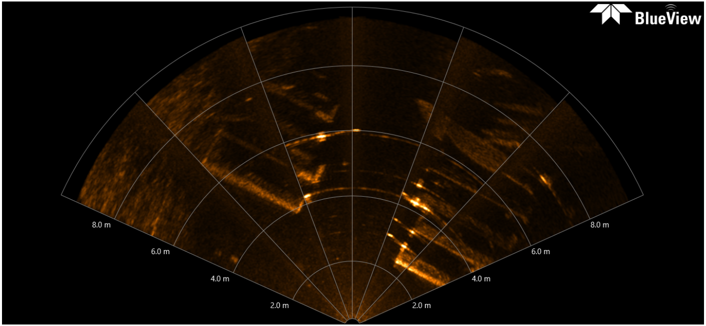
Convergence is solvable in ASYNC for ϵ and θ sufficiently small.

First Conclusion

- This is fun!
- Results also for pattern formation, covering, and flocking
- Also many results for robots on graphs

Example Mission 1: Inspection

- swarm of 5 AUVs; unstructured; no leader
- inspect subsea cable or pipe
- before mission: all AUV have complete information
- once deployed: limited visibility; communicate via sonar
- autonomously navigate to cable; collect data; resurface
- example challenge: reconfigure to replace malfunctioning AUV



Example Mission 2: Exploration

- swarm of 10 AUVs; unstructured; no leader; other AUVs in reserve
- explore unknown underwater area
- look out for singular points
- when AUV detects singular point: remain above; tell others
- others reconfigure and continue search
- when second point detected: first AUV released; rejoins swarm
- AUVs continually replaced (autonomy 10 hours)

Conclusion

- Both theoretically and practically, distributed robotics is fun
- Nice theoretical results which have never been tested in practice
  (*« Un robot peut-il suivre un autre ? »*)
- Need more theory; need to test theory in practice
- Visions of (practical) distributed underwater robotics rather far removed from theory

Third International Workshop on

Methods and Tools for Distributed Hybrid Systems

Amsterdam, Netherlands, 26 August 2019

Associated with CONCUR 2019

The purpose of DHS is to connect researchers working in *real-time* systems, *hybrid* systems, *control* theory, *distributed* computing, and *concurrency*, in order to advance the subject of **distributed hybrid systems**.

Distributed hybrid systems, or distributed *cyber-physical* systems, are abundant. Many of them are safety-critical, but ensuring their correct functioning is very difficult. We believe that new techniques are needed for the analysis and validation of DHS. More precisely, we believe that convergence and interaction of methods and tools from different areas of *computer science*, *engineering*, and *mathematics* is needed in order to advance the subject.

The first DHS workshop was held in Aalborg,

Invited Speakers

Thierry Grousset
Kopadia, Paris
France

Majid Zamani
University of Colorado Boulder
United States

Xavier Urbain
Université Lyon 1
France