The Cooperative Cleaners Case Study: Modelling and Analysis in Real-Time ABS

Silvia Lizeth Tapia Tarifa

Precise Modelling and Analysis
University of Oslo

sltarifa@ifi.uio.no

29.11.2013
Outline

1 Motivation
   - Swarm Robotics
   - ABS
   - Problem Statement

2 The Cooperative Cleaners Case Study
   - Definition
   - The Floor
   - The Cleaning Robots
   - The CLEAN Protocol

3 Results
   - The Cooperative Cleaners: Model in Real-Time ABS
   - The Cooperative Cleaners: Analysis and Simulations

4 Conclusions

5 Future Work
Swarm Robotics

Group of **autonomous, simple and similar robots** is coordinated in a way that leads to **useful behaviour** of the swarm itself.

**Advantages**

- **Parallelism**: robots cooperate and synchronise intelligently.
- **Decentralised**: robots operate on local information to accomplish global goals.
- **Adaptability**: changing environment.
- **Scalability**: e.g., number of robots.
- **Fault-tolerance**: robots complete the task, even if some of them fail.
ABS – Abstract Behavioural Specification Language

ABS is a language for modelling object-oriented systems at an abstract, yet precise level.

- Targets software systems that are: concurrent, distributed and object-oriented.
- Offers a wide variety of modelling options in one framework. (datatypes, functions, classes, concurrency, distribution, etc.).
- Clear and simple concurrency model that permits synchronous as well as asynchronous communication.
- Abstracts from implementation choices of data structures.
- Fully executable with code generators into e.g., Java and Maude.
- Has a formal semantics.

Real-Time ABS extends the syntax and semantics of ABS in order to allow models with time dependent behaviour.
About This Thesis

Questions:

1. How can Real-Time ABS be used to naturally model autonomous, decentralised and self-organised systems such as swarm robotics?

2. To what extent can the simulation tool of Real-Time ABS help to analyse the collective behaviour of such systems?

Approach:

- Develop a case study from the swarm robotics domain that exploits features of Real-Time ABS such as concurrency, distribution, object-orientation and so forth.
- Evaluate how well Real-Time ABS can be applied to model and analyse this case study.
The Cooperative Cleaners Case Study

A group of cleaning robots cooperate to clean a dirty floor

- **Floor:** connected dirty area in $\mathbb{Z}^2$.

- **Cleaner:**
  - restricted amount of memory,
  - indirect communication (signals and sensing),
  - local knowledge (e.g., overall topology of the dirty floor is unknown),
  - movement follows a protocol.

Cleaning the whole floor is an emerging property of the cleaning robots cooperation

Definition of the Problem: The Floor

- **Shape:** undirected graph $G$ in $\mathbb{Z}^2$ with vertices $v = (x, y)$ representing positions connected with edges $e = (v, w)$ following a 4-neighbours relation.

- **Concepts:** 4-neighbours, 8-neighbours, boundary, single connected component, critical vertex.

- The **dirty floor** $F_t$ is a subgraph of $G$, $t$ represents time.

- **Initial state:** assume $G$ is a single connected component without holes or obstacles, and $F_0 = G$. 
Definition of the Problem: The Cleaners

- Group of identical cleaners moving across $F_t$.
  (using 4-neighbours relation).

- Cleaners can only observe their local environment.

- **Goal**: clean the dirty floor
  (not prior knowledge of the shape or size of the dirty floor).

- All the cleaning robots start and stop in the same position.

- **Initial position**: at the boundary of $F_0$. 
Definition of the Problem: The CLEAN Protocol

- **Common protocol:** moving and cleaning (non-critical positions) along the boundary of $F_t$.
- **Cyclic algorithm:** in each discrete time step, each cleaner executes one outer cycle.
- The cleaners “peel” layers from the boundary of $F_t$, until $F_t$ is cleaned entirely.
- **Protocol must preserve the connectivity** of the dirty floor.

From: Cooperative cleaners: A study in ant robotics. Wagner et al.

S. Lizeth Tapia Tarifa
The Cooperative Cleaners Case Study
1. User-defined datatypes and their associated functions.

- To represent and manipulate the information of the floor and the cleaners.
- To record monitoring information.

```java
type Pos = Pair<Int, Int>;

type Graph = Set<Pos>;

... 

type CleanersPerPos = Map <Pos,Cleaners>;

...

type FloorPath = List<Triple<Time,Pos,String>>;

...

type CleanerPath = List<Pair<Time,Pos>>;

... 
```
2. Abstracting hardware of cleaners and environment.

- Abstracted using method calls.
- Modelling entity environment.
- Passive floor vs. reactive environment.
3. Interpretation and implementation of informal concepts as functions.

```
def PosSet eightNbr(Pos p, Graph g) =
  let (Pos u)=up(p) in
  let (Pos ul)=upleft(p) in
  let (Pos ur)=upright(p) in
  let (Pos l)=left(p) in
  let (Pos r)=right(p) in
  let (Pos d)=down(p) in
  let (Pos dl)=downleft(p) in
  let (Pos dr)=downright(p) in
  dirtySet(set[u,ul,ur,l,r,d,dl,dr],g);
```

```
def PosSet dirtySet ...
```
4. Ambiguities of the CLEAN protocol.

- Unexplained “Check Near Completion of Mission” subtask.
  - SWEEP Protocol.
- Adaptation of the “Calculate waiting dependencies” subtask.
- Implicit synchronisation: we used fixed timers to guarantee homogeneous progress. We restricted the execution of the CLEAN protocol to only one cycle per time step.
4. Ambiguities of the CLEAN protocol.
During the modelling process:

1. Validate that the manipulation of user-defined datatypes is correct.
2. Validate that the implementation of the concepts given in the problem description is adequate.
3. Observe that the recording of the monitoring information was consistent.
4. Observe that the model satisfies the considered safety properties.
The Cooperative Cleaners: Analysis and Simulations

Cleaning a five per five square floor using one cleaner

Path of the cleaner

Progress of cleaning the floor

Total time: 36
Cleaning a five per five square floor using two cleaners

Total time: 31
The Cooperative Cleaners: Analysis and Simulations

Cleaning a five per five square floor using three cleaners

S. Lizeth Tapia Tarifa
### Summary of simulation results.

<table>
<thead>
<tr>
<th>Size</th>
<th>Cleaner Count</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>5x5 (size = 25)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 cleaner</td>
<td></td>
<td>36</td>
</tr>
<tr>
<td>2 cleaners</td>
<td></td>
<td>31</td>
</tr>
<tr>
<td>3 cleaners</td>
<td>(before livelock)</td>
<td>23</td>
</tr>
<tr>
<td><strong>10x10 (size = 100)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 cleaner</td>
<td></td>
<td>140</td>
</tr>
<tr>
<td>2 cleaners</td>
<td></td>
<td>129</td>
</tr>
<tr>
<td>3 cleaners</td>
<td>(before livelock)</td>
<td>83</td>
</tr>
<tr>
<td><strong>20x20 (size = 400)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 cleaner</td>
<td></td>
<td>580</td>
</tr>
<tr>
<td>2 cleaners</td>
<td></td>
<td>587</td>
</tr>
</tbody>
</table>

S. Lizeth Tapia Tarifa
Q₁. How can Real-Time ABS be used to naturally model autonomous, decentralised and self-organised systems such as swarm robotics?
Conclusions

Real-Time ABS & modelling of swarm robotic systems

- Special hardware. Entity *environment*.
- Passive environment. Entity *environment*.
- Implicit communication. Entity *environment*.
- State representation. User-defined datatypes.
- Work flow. Functional + Imperative.
- Monitoring information. User-defined datatypes.
- Synchronisation. Modelling a suitable synchronisation mechanism for a given concurrent problem represents a challenge on its own (e.g., fixed timers).
Q₁. How can Real-Time ABS be used to naturally model autonomous, decentralised and self-organised systems such as swarm robotics?

A₁. From the cooperative cleaners case study:
- almost naturally model in Real-Time ABS (simple concurrent model, functional layer, imperative layer),
- it required fairly advanced modelling skills.
Conclusions

Q$_2$. To what extent can the simulation tool of Real-Time ABS help to analyse the collective behaviour of such systems?
RTABS & analysis of swarm robotic systems

- **Simulations.**
  - Useful while modelling the case study.
  - Limitations with respect to handling large amount of data.

- **Analysis of monitoring information.**
  - Insights about the behaviour of the system.
  - Manual analysis.
  - Output format from the tool is not user friendly.
  - Hard to scale in the size of the floor and in the number of cleaners.
Conclusions

Q2. To what extent can the simulation tool of Real-Time ABS help to analyse the collective behaviour of such systems?

A2. From the cooperative cleaners case study:

- can be analysed using simulations,
- restricted analysis to small scenarios,
- requires reformatting and reorganisation of obtained monitoring data.
Future Work

- Possible changes to the model and to the protocol itself.
  - Group of robots with coordinators.
  - Different synchronisation mechanisms (e.g., flexible timers).

- Possible extensions to the analysis, using symbolic execution.
  - Reasoning about invariants using KeY.
  - Reasoning about cost analysis using COSTA.
THANK YOU