Collective Robotics

IAR Lecture 12

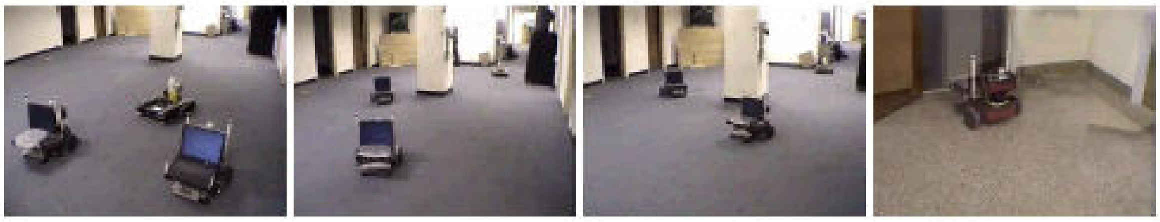
Barbara Webb

Collective robotics

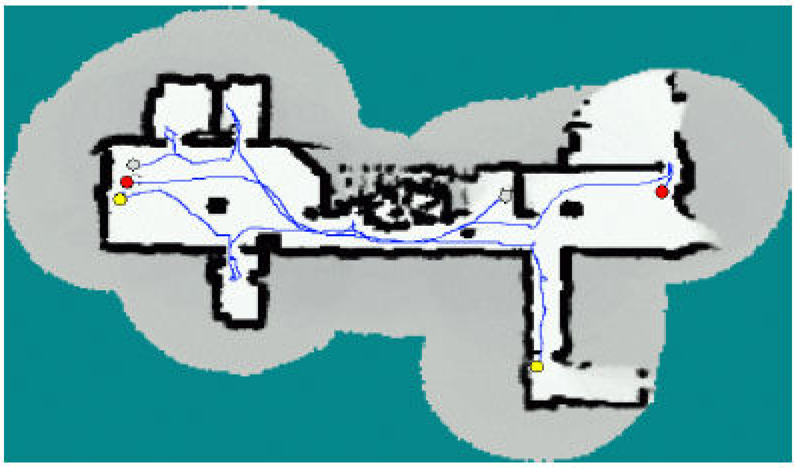
Different approaches (Kornbach, 2013)

- **Co-operative:** distributed sensing and actuation, but centralised control.
- **Networked:** higher individual autonomy, but still high level of communication and common knowledge.
- **Swarm:** no common knowledge, only local communication, or interaction via effects on the world.
- **Small world:** minimalist capabilities of individual, collective computation.

E.g. exploring with multiple robots



- Provided robots can merge their maps, can explore faster with multiple robots.
- Potential speed up is 2*k, as single robot would need to spend time traversing known space to get to new frontier.



- But need to co-ordinate exploration.

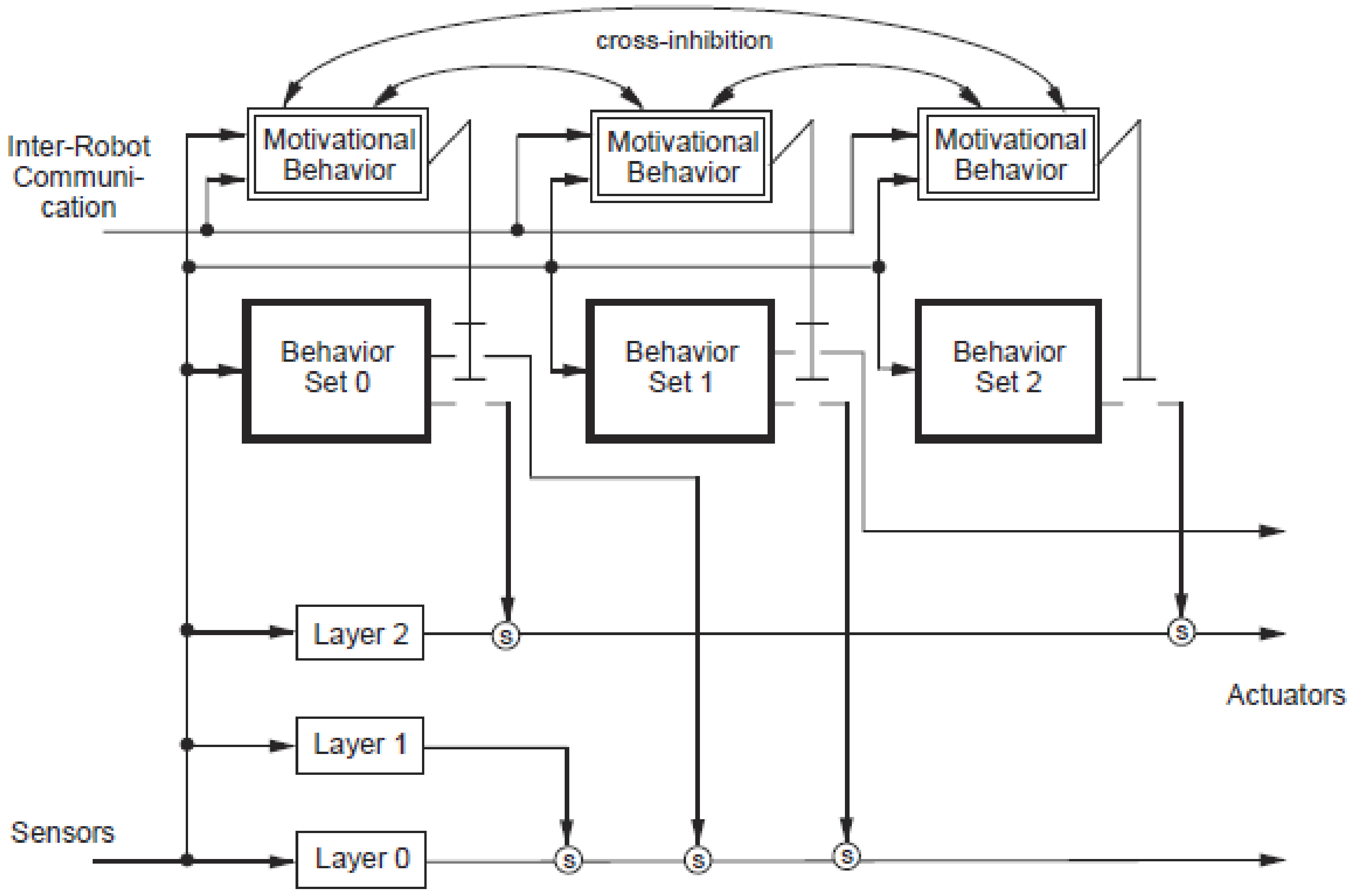
Active mapping for k robots

  - For each robot, obtain cost function of moving from current position to other possible grid positions.
  - Use binary representation of potential information gain: cell is 0 if explored, 1 if unexplored.
  - For each robot in turn, set goal as unexplored grid location that has minimum cost to reach, and mark that location as explored (so not available for next robot).
- More sophisticated approach would allow robots to swap goals if this reduces the overall cost, e.g. using auction mechanism.
- Generalised, this is the problem of task allocation.

E.g. Behaviour based task allocation

- ALLIANCE architecture (Parker,1998)
  - Robots have motivation systems determining action selection:
    - Impatience: will choose a task not being completed by other robots
    - Acquiescence: give up a task if failing to complete
  - Broadcast periodic messages to each other indicating what they are doing.
  - Sensory feedback to monitor progress on tasks.

The ALLIANCE Architecture



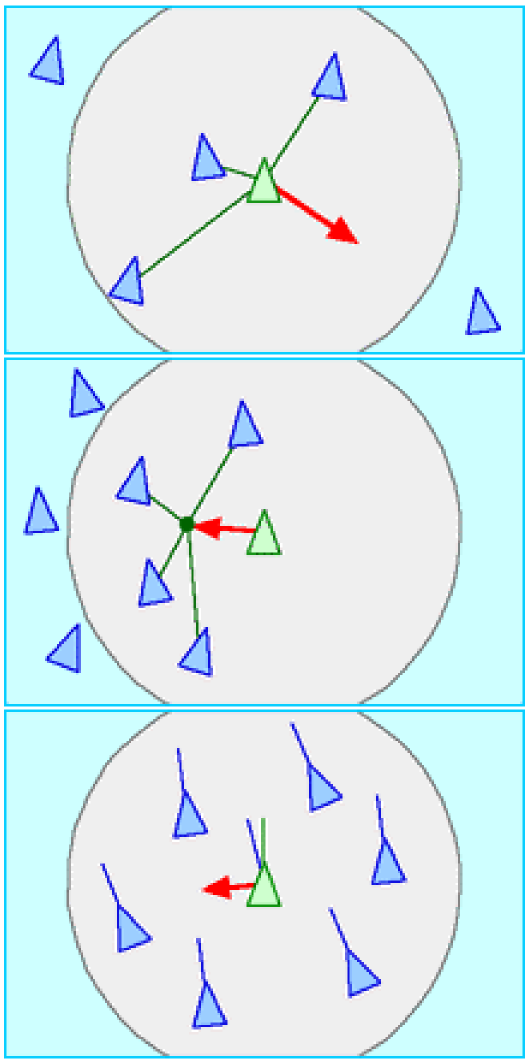
Emergent co-operation

- Holland (1995) ‘Stigmergy’
- Robots:
  - Front scoop tends to collect pucks
  - Lever triggers switch if pushing two or more, makes robot back up, leaving pucks behind
  - Also avoid walls and each other using IR.
- Result is gradual aggregation of pucks in a single pile

Emergent co-operation

- Melhuish et al (2000)
- Robots can carry puck, detect gradient and notice if cross a boundary line.
- Simple rules:
  - If hit another object, drop puck
  - If cross boundary going up gradient, move short distance and drop puck
  - If cross boundary going down gradient, back-up short distance and drop puck

Emergent group behaviour



Flocking: (Reynolds 1987)

Assumes all ‘boids’ are identical and follow the same local rules:

1. **Collision Avoidance:** Separate from other boids.
2. **Centering:** Stay close to other boids.
3. **Velocity matching:** travel in same direction.

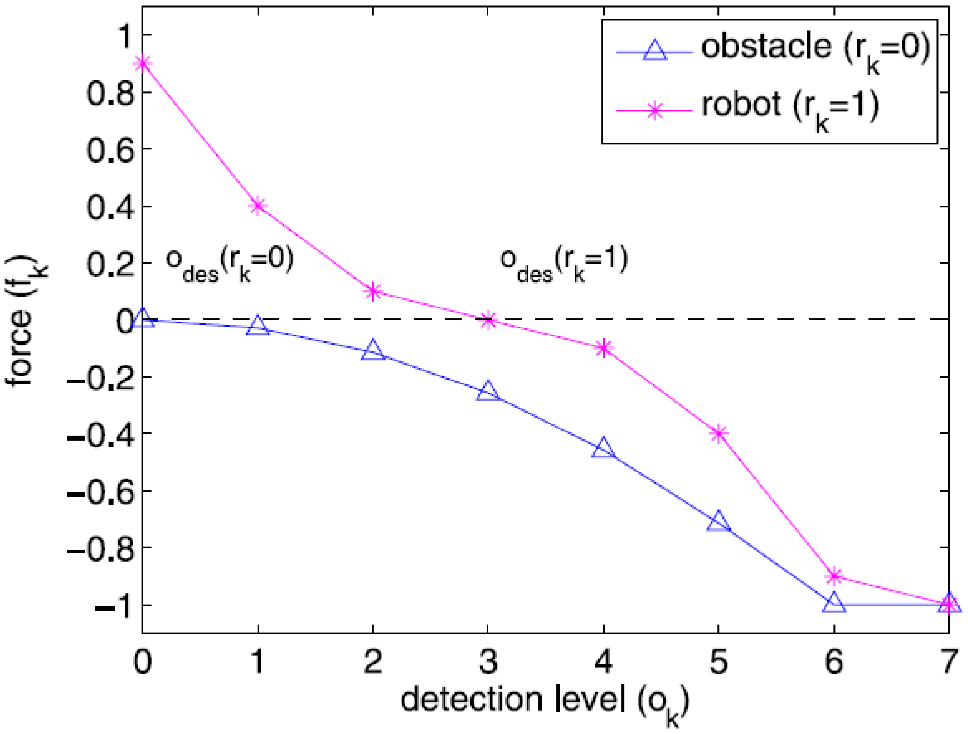
Emergent group behaviour

Flocking in real robots:

- various attempts, but needed to include virtual or explicit leader, or all robots sensing goal
- also problem of how to make individual robot able to sense relative position and bearing of neighbours
- Recent example addresses some of these limitations (Turgut et al., 2008)

Turgut et al. 2008

- IR sensors used to detect other robots (when not active) and obstacles (when active)
- Use virtual heading sensor: each robot has a compass and wirelessly broadcasts its direction to neighbouring robots.
- Desired heading alignment is calculated as average of detected neighbours
- Proximal control is calculated as a virtual force with respect to detected robots or obstacles



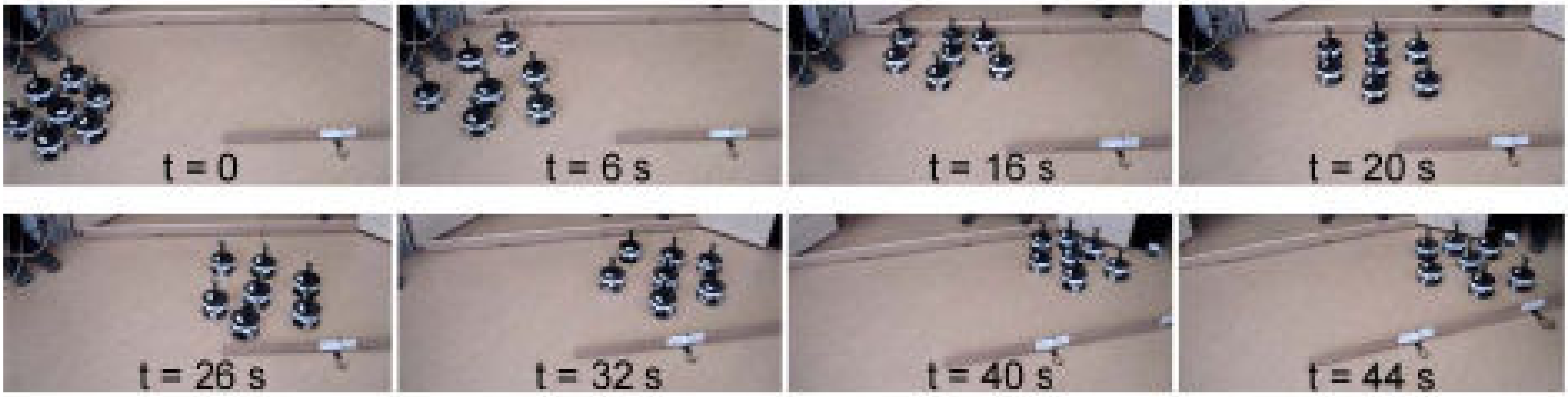
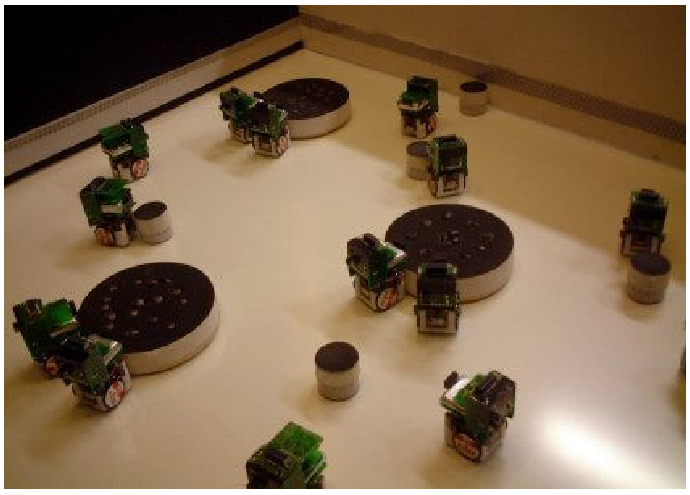


Fig. 14 Self-organized flocking with seven Kobots. Starting from a connected but unaligned state, Kobots negotiate a common heading and move as a group in a constrained environment and bounce off a wall without losing their cohesion

- Is robust to noise, but find coherent swarm size depends on virtual heading sensor range: noise in system prevents long-range order emerging from short-range interactions.
- Large flocks possible with just a few long-range interactions or with some common homing information.

- Basic task allocation assumes all robots can do all tasks and just need to distribute them effectively, then work separately.
- More complex scenarios:
  - Task might require complex continuous interaction between two or more robots.
  - Robots could be heterogeneous.
  - Robots could be interacting with other technologies, or humans (or animals).



Inverse flocking: modelling duck behaviour by simple flocking rules to produce sheepdog control algorithm (Vaughan et al 2000)

Reconfigurable robots

- Superbots
- MTRAN3
- MBlocks

References

Serge Kernbach (2013) Handbook of Collective Robotics. CRC Press.

Lynne Parker (1998) *ALLIANCE: An Architecture for Fault Tolerant Multi-Robot Cooperation* IEEE Transactions on Robotics and Autonomous Systems, 14(2):220-240

Owen Holland and Chris Melhuish (1999) *Stigmergy, self-organization, and sorting in collective robotics* Artificial Life 5:173 - 202

Craig W.Reynolds (1987) *Flocks, Herds, and Schools: A Distributed Behavioral Model*, in Computer Graphics, 21(4) (SIGGRAPH '87 Conference Proceedings) pages 25-34. (http://www.red3d.com/cwr/papers/1987/boids.html)

Turgut, A.E., Celikkanat, H., Gokce, F. and Sahin, E. (2008) Self organized flocking in mobile robot swarms. Swarm Intelligence 2: 97-120

Vaughan, R., Sumpter, N., Henderson, J., Frost, A. and Cameron, S. *Experiments in Automatic Flock Control* (2000) Journal of Robotics and Autonomous Systems 31 pp.109-117