Published in Autonomous Robots for Research and Edutainment - AMiRE 2001, Proceedings of the 5th International Heinz Nixdorf Symposium, pp. 253–260, Paderborn, Germany, October 2001

Performance Analysis of a Colony of Locally Communicating Robots

B. Iske and U. Rückert

System and Circuit Technology, Heinz Nixdorf Institute,
Paderborn University, Fürstenallee 11, D-33102 Paderborn

E-mail: `iske@hni.uni-paderborn.de`

Abstract. A cube clustering problem which is to be solved by a colony of locally communicating robots is introduced. The performance of the robot colony solving the cube clustering problem is analysed regarding the number of £nal cluster points and the optimal robot density that solves the problem the fastest. Parallel computing theory is applied to determine the optimal robot density. The modelled performance matches with experiments executed with the minirobot Khepera.

1 Introduction

Research in multi-robot systems has grown signi£cantly in recent years. Through cooperation multi-robot systems are expected to solve tasks faster or those more complicated than single robots are able to. Thus, the variety of multi-robot research ranges from communication analysis over task planning and exploration to learning and object transportation [4].

This paper concentrates on the synergy effects of locally communicating robots and how a colony of them can work together effectively while maintaining their individual independency. What we consider a colony of robots to be is a large number of robots in the order of tens to hundreds. This term is borrowed from biology where a colony of ants or bees consists of even larger orders of individuals. All robots work independently on the same task and in cases where a single robot fails or breaks down the others should still remain working. Stigmergy is the technical term that describes cooperation within colonies of individuals without a global coordination [6].

Section 2 of this paper describes the cooperative cube clustering problem (CCC problem). It also includes a strategy analysis of the determination and distribution of a common coordinate system to all individuals of the robot colony. Furthermore, the minirobot Khepera as an individual of the colony is introduced, and a comparison between a colony of robots and parallel computing is drawn. A model of a colony of locally communicating robots is established and analysed concerning optimal robot density in section 3.

2 The Cooperative Cube Clustering Problem

The main objective with the CCC problem is the development of a flexible and redundant robot colony system. Flexible in this case means robustness against failure of single robots as well as the possibility of adding an arbitrary number of new individuals during running time. To achieve redundancy all robots are identical which means that there is no master and the individual robots cannot be distinguished. Only this ensures that in case of failure of single robots the overall system remains running. Having identical robots also allows the adding of new robots afterwards.

The task to be analysed here is a simple cube clustering problem. Cubes are randomly distributed across a plane surrounded by walls and a colony of robots is to cluster the cubes through pushing to one global cluster point. In order to ensure all robots have the same cluster point they need to have a common coordinate system which is achieved through communication among the robots. This problem is an extension of the cube clustering presented in [3] since it includes data exchange and communication.

This simple example of cube clustering poses interesting questions concerning robot colonies. How can data be distributed among robots, how can coordinate systems be adjusted and how many robots should be used to gain best performance?

2.1 A strategy for determining and distributing the common coordinate system

To achieve the goal of only one final cluster point all robots need to have the same coordinate system. For maintaining independency of the environment no landmarks should be used. Therefore, each individual robot starts with its own relative coordinate system. The first box found by a robot is set to be the origin of its coordinate system where upon all boxes found by the robot are pushed to this point. Assuming the number of robots to be N there would be a maximum of N clusters of boxes without the exchange of coordinate systems.

Adjusting the coordinate systems among robots is a special data exchange problem since not only data needs to be exchanged but also to relative position of the communication partner needs to be known. This is why a global communication system is not suitable for such a problem. A global communication system would also reduce the flexibility of the system (adding / removing of robots and independency of the environment) and require more resources (sending power and bandwidth). Therefore, a local communication system is used where robots only exchange data in case they meet one another. The advantage of local communication upon meeting is that the relative position of the communication partner is known and the coordinate systems can be adjusted easily. The main disadvantage is that not all robots get the information at the same time consequently. With local communication the information exchange is similar to a diffusion process. A spatial and temporal analysis of the information diffusion in a robot colony can be found in [7].

The most promising strategy for determining the coordinate system that is made common to all robots is based on age. Each robot knows the age of its own cluster point. If a robot meets another the ages of their cluster points are exchanged and the coordinate system of the robot with the younger cluster point is adjusted to that of the robot with the older cluster point. The first advantage of this strategy is that it is more likely that more boxes are already clustered at the older cluster point. Secondly, only one cluster point should ideally exist in the end. Since there is no global communication, there is no appropriate strategy which can achieve a single £nal cluster point. However, the probability that two or more cluster points remain can be minimised. With this ‘based upon age’ strategy, the probability of exactly one £nal cluster point is the same as the probability that the age of all robots’ cluster points is different. This can be calculated as the ratio of an unordered random sample without replacement (number of possibilities of all cluster points having different ages) and an unordered random sample with replacement (number of possibilities of cluster points having any age). Let N be the total number of robots initially in the environment and t the maximum duration the robots require to £nd the £rst box. With the further assumptions, that the time it takes a robot to £nd the £rst box is equally distributed across t, and the robots can measure time with a resolution of Δt, which results in $M = \frac{t}{\Delta t}$ distinguishable points in time, the probability P of only one cluster point can be calculated as follows:

$$P = \frac{\frac{M!}{(M-N)!N!}}{\frac{(N+M-1)!}{N!(M-1)!}} = \frac{M!(M-1)!}{(M-N)!(M+N-1)!} \tag{1}$$

Realistic values for the number of robots range from 2 to 50 with the number of distinguishable points in time being in the order of thousands since it takes up to a couple of seconds until a robot, which measures time in milliseconds, £nds its £rst box. The probability of only one cluster point is dependant on the number of robots and the number of distinguishable points in time. This is shown in Figure 1.

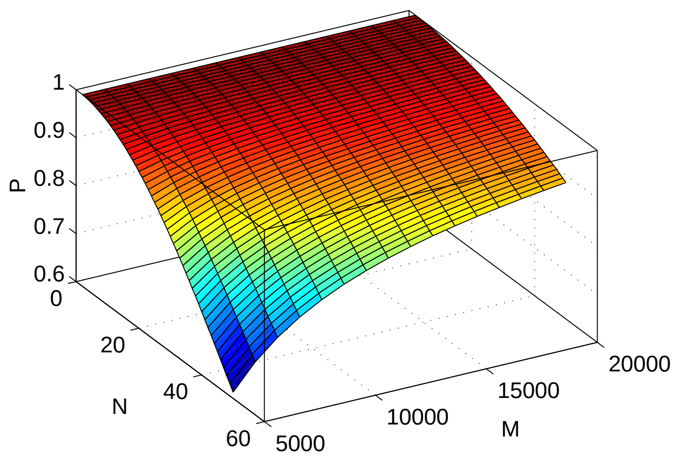

Figure 1. The probability P for only one £nal cluster point with a coordinate system adjustment strategy based upon age. The probability is dependant on the number of robots N and the number of distinguishable points in time M.

2.2 Comparing a colony of locally communicating robots to parallel computing

The solution of the CCC problem that uses a colony of robots instead of only a single robot resembles the approach followed in parallel computing. Instead of having one processor solving a problem serially several processors are employed to solve the problem in parallel and thus faster. Amdahl’s Law is a law governing the speedup of using parallel processors on a problem, versus only one serial processor. Speedup is de£ned as the time it takes a problem to be solved serially (with one processor) divided by the time it takes to solve the problem in parallel (with N processors). The formula for speedup is [2]:

$$S = \frac{T(1)}{T(N)} \tag{2}$$

If K is the percentage of the problem, which is strictly serial, and N the number of processors used, the speedup can be calculated as follows:

$$S = \frac{K \cdot T(1) + (1-K) \cdot T(1)}{(1-K) \cdot T(1)/N + K \cdot T(1)} = \frac{N}{(1-K) + N \cdot K} \tag{3}$$

Another measure often used in parallel computing is ef£ciency. It measures the fraction of time for which a processor is usefully employed. It is de£ned as the ratio of speedup to the number of processors [2]:

$$E = \frac{S}{N} \tag{4}$$

The approach described for the CCC problem does not contain a strictly serial part. However, the communication between the robots can still be regarded as serial since it decreases the robots' effective working time.

The communication process includes the following major distinctive points between the CCC problem and parallel computing:

- The frequency of communication is increasing with the density of robots
- The communication volume per communication remains constant
- Communication is with random partners only
- Communication happens only at random points in time

Despite the differences, Eq. 3 can be applied to the CCC problem and the speedup using a colony of robots can be evaluated. The trade-off between speedup through a high density of robots working in parallel and the fact that a higher density of robots results in more frequent communication (a higher percentage of the serial part) can thus be optimised.

2.3 The Khepera minirobot as an individual of the robot colony

The Khepera minirobot is an autonomous system often used in research [5]. In this case several of them are used as individuals in the robot colony. In its base con£guration the Khepera is equipped with a processor, memory, a battery, two DC motors for locomotion and eight proximity infrared sensors for environment recognition. The infrared sensors are active, which means that each of them is sending an impulse and detecting obstacles by measuring the intensity of the re¤ected light. Therefore, besides recognition they can also be used for communication by sending an impulse and reading the sensor value separately. Due to hardware restrictions the robot can access only one active sensor at a time. This means the robot can either read the value of a certain sensor or send a signal via a certain active sensor. The two front sensors of the Khepera are set as the sending and receiving sensor respectively. This is the reason why the robots have to justify in order to be able to communicate. The justifying procedure is implemented as a simple feed forward network based on 'braitenberg's' vehicle [1]. Due to noise and

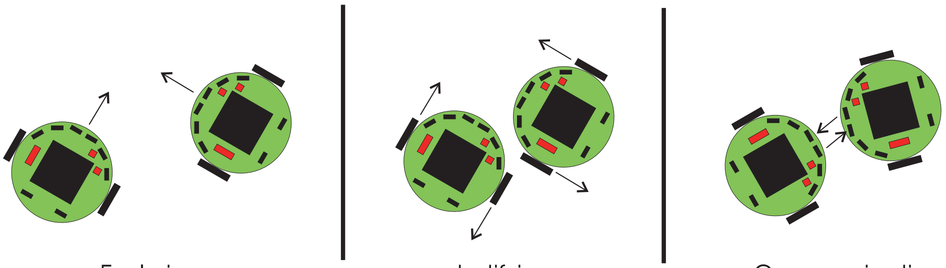

Figure 2. Two robots are exploring the environment. If a robot detects an obstacle it justi£es. In this case the obstacle is another robot which also justi£es. After justi£cation both robots start a synchronisation process which succeeds if the obstacle is another robot. After synchronisation the two robots start communicating.

interference with the impulses of the opposing robot, the medium angular misalignment after justi£cation is around 8° and the distance error is a maximum of 1cm. Errors can therefore be neglected. The justifying procedure is shown in Figure 2.

For communication a NRZ-Signal (Non Return to Zero) encoding is used in conjunction with a serial protocol—one start bit, eight data bit, and one parity bit. The net datarate is 222bit/s with a communication distance of a maximum of 3cm. Considering that the Khepera proximity sensors are converted to communication devices, the datarate and communication distance are in the expected range.

3 The Local Communication Model

In this section a model of a colony of locally communicating robots, which is used to solve the trade-off between a high density of robots to enhance parallel working and a low density of robots to enhance parallelism (decrease the serial portion of the problem by reducing communication) is introduced. The model approximates the local communication of Khepera robots and does not raise the claim of being exact. It should rather be seen as a guideline for local communication in robot colonies and with minor modi£cations it can be adapted to other locally communicating systems.

The robots randomly search the environment in order to £nd boxes. Pushing the boxes to the cluster point is approximated as a random walk of the environment as well. With this precondition and denoting the density of robots with ρ the probability of N robots existing in area A can be modelled Poisson distributed as follows [7]:

$$P(\rho, N) = \frac{(\rho A)^N}{N!} e^{-\rho A} \tag{5}$$

The major simpli£cation of this approach is that the probability of robots existing in area A is assumed to be independent of the robots existing in another area and that the distribution of robots is independent of different areas of the environment.

In order to evaluate the average frequency of communication some further variables are introduced:

A_C:	communication area, de£ned through the maximum area where a robot can recognise another and start communicating
A_E:	area of the whole environment
t_{step}:	average time it takes a robot to cross A_C
t_{com}:	duration of one local communication
t_l:	lifetime of a robot (identical for all robots)

Since a probability theory approach has to be employed the duration of one sample needs to be speci£ed and is here set to t_{step}. The effective working time t_{eff} of robot x is the portion of the robot's lifetime where no other robot remains in robot x's communication area. If exactly one other robot enters robot x's communication area, communication which lasts for t_{com} seconds starts. With ρ designating the number of robots per communication area excluding robot x ($\rho = (N-1)A_C/A_E$), the effective working time of a single robot is given as:

$$t_{eff(\rho,t_{com})} = \frac{t_l}{t_{step}P(\rho,0) + t_{step}\left[1 - P(\rho,0) - P(\rho,1)\right] + \left[t_{com} + t_{step}\right]P(\rho,1)} t_{step}P(\rho,0) \qquad (6)$$

Figure 3 shows the effective working time of one robot in a colony of $N = \rho A_E/A_C + 1$ robots. The effective working time is independant of the environment size. A realistic value for the robot's lifetime t_l is 45 minutes and for t_{step} 1.5s are measured.

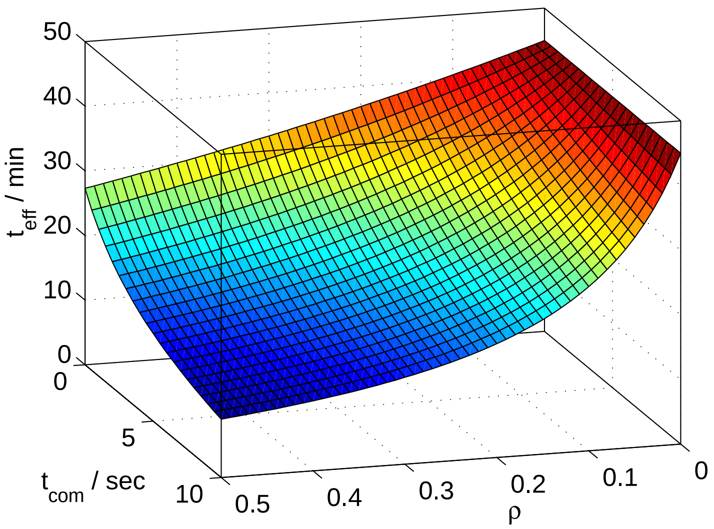

Figure 3. The effective working time of one robot. In this case using only one robot yields the highest ef£cient working time, since a higher density of robots results in communication, which is absent in case of only one robot. A longer duration of communication lowers the effective working time.

3.1 Speedup and Ef£ciency

To calculate the speedup, the percentage K of the problem, which is strictly serial needs to be known. With the effective working time, K can be calculated as $K = 1 - \frac{t_{eff}}{t_l}$. The speedup and ef£ciency of employing a colony of robots using local communication is shown in Figure 4.

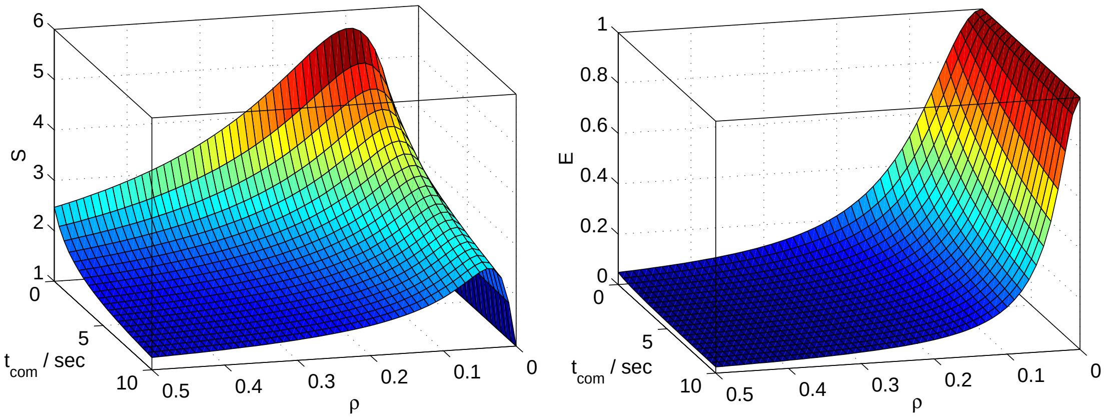

Figure 4. The two plots show the speedup S and ef£ciency E of a colony of locally communicating robots. The environment size is set to $A_E = 100 \cdot A_C$. Low robot densities gain the best speedup with ef£ciency falling greatly with increasing robot density.

3.2 Optimal Robot Density

To solve the CCC problem in the shortest time, the optimal robot density where the speedup is maximal needs to be calculated. The optimal robot density not only depends on the duration of communication but also on the size of the environment since the number of robots N, which is a factor in the speedup formula, increases with the size of the environment.

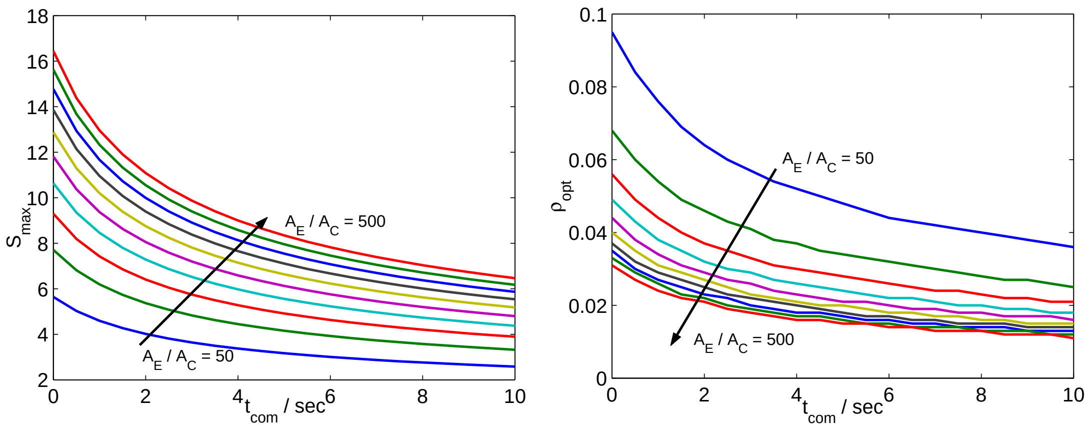

Figure 5. The two plots show the maximal speedup S_{max} and the optimal robot density ρ_{opt} depending on the communication duration and the size of the environment. For increasing environment size the speedup increases since more robots can be employed whereas the optimal robot density decreases.

Whereas one would expect a constant optimal robot density independent of the environment size Figure 5 shows that the optimal robot density decreases with the increasing size of the environment. This is because for constant robot density the number of robots increases with increasing environment size, and the maximum speedup for increasing robot numbers is gained for a decreasing serial part, which means a lower robot density. Considering Eq. 7 one can see that the maximal speedup for an in£nite area is obtained for $K \to 0$, which corresponds to $\rho \to 0$.

$$\lim_{A_E \to \infty} S = \lim_{A_E \to \infty} \frac{\rho A_E / A_C + 1}{(1-K) + (\rho A_E / A_C + 1) \cdot K} = \frac{1}{K} \tag{7}$$

3.3 Experimental Results

In the CCC problem the khepera robots have to justify before they can start communicating with a datarate of only 222bit/s. After communication the robots turn away from each other resulting in a communication procedure of about 9s. Experiments showed that indeed the optimal robot density is rather low. In an environment size of 91cm x 91cm and a local communication area of $A_C = 154\text{cm}^2$ ($A_E \approx 54 \cdot A_C$), best results concerning the effective working time were achieved with 3 robots, which corresponds to $\rho = 0.037$. The calculated optimal robot density for $A_E = 50 \cdot A_C$ is $\rho = 0.037$ and approximates the measured results very well.

4 Summary and Discussion

In this paper we modelled the effective working time of independent robots in a colony of locally communicating robots. Given the effective working time, parallel computing theory is applied to compute the optimal robot density. The result is that for larger environments the optimal robot density decreases. The model could be validated with an implementation of the CCC problem on the minirobot Khepera.

One problem with the optimal robot density is that the density is very low which means that robots meet rather rarely. Since local communication is employed to distribute data among many robots it may take quite long until data is distributed to all robots. In order to achieve an optimal overall performance, which means a high effective working time and at the same time fast distribution of data, the temporal and spatial distribution of data in locally communicating robot colonies studied by Yoshida [7] has to be considered as well.

References

[1] **V. Braitenberg.** Vehicles. MIT Press: Cambridge (Mass.), 1984.

[2] **V. Kumar, A. Grama, A. Gupta, and G. Karypis.** Introduction to Parallel Computing. The Benjamin/Cummings Puplishing Company, Inc., 1994.

[3] **A. Löf¤er, F. Mondada, and U. Rückert.** Experiments with the Mini-Robot Khepera. *Proceedings of the 1st International Khepera Workshop*, December 1999.

[4] **L. E. Parker.** Current State of the Art in Distributed Autonomous Mobile Robotics. *5th Intern. Symposium on Distributed Autonomous Robotic Systems*, 2000.

[5] **K-Team, S. A.** Robots. http://www.k-team.com/robots/khepera/khepera.html, 2001

[6] **G. Theraulaz and E. Bonabeau.** A Brief History of Stigmergy. *Arti£cial Life* No. 5, pp 97-116, 1999.

[7] **E. Yoshida and T. Arai.** Performance Analysis of Local Communication by Cooperating Mobile Robots. *IEICE Transactions on Communication* Vol.E83-B, No. 5, pp 1048-1059, May 2000.