Proceedings of the 2006 IEEE
International Conference on Control Applications
Munich, Germany, October 4-6, 2006

Control of the Multi Agent Micro-Robotic Platform MiCRoN

Panagiotis Vartholomeos, Savvas Loizou, Michael Thiel, Kostas Kyriakopoulos and Evangelos Papadopoulos

Abstract—**This paper presents the theoretical framework for the centralized control architecture of the multi agent micro-robotic platform MiCRoN. The entire control system architecture integrates sensory modules, modeling modules, and control modules. The latter are composed by (i) a high level simulation and autonomous execution unit that is capable for on-line multi-robot navigation with collision avoidance, (ii) a trajectory tracking unit for manipulation purposes, and (iii) a low level position controller that performs position control exploiting machine learning algorithms. The high level controllers take into account behaviors specific to the micro-scale. The performance of the layered control system is evaluated through simulations and preliminary hardware experiments on a micro-robotic platform. The application domain of the MiCRoN platform is cell manipulation, and 3-D assembly for micro-fabrication.**

I. Introduction

THE increasing scientific and technological advancements in the domain of biotechnology and nano-technology demand for simultaneous progress on the domain of micro and nano-robotics. Especially, micro and nano manipulation/assembly will play major roles in the processing of micro and nano entities such as biological cells [1], DNAs, neurons [2], nanotubes and nanostructures [3]. Several promising micro-actuation techniques have been devised in the last decade allowing detection and manipulation of micro and nano-objects [4-6]. Furthermore, motion mechanisms and micro-robotic platforms have been designed that allow for flexible robotic systems capable for speeds up to several mm/s and for nanometric manipulation precision [7-9]. However, most of the research efforts in this field are still restricted to the accomplishment of non-cooperative tasks with reduced complexity. Also in most of the cases the processes have poor-repeatability and low reliability, which do not allow for systematic micro nano-manipulation and assembly with high throughput.

Manuscript received February in 10/ 02/ 2006. This work was supported by the Europian Union through contracts: MICRON (IST-2001-33567), I-SWARM (IST-2004-507006).

P. V. Author is with the National Technical University of Athens, Mechanical Engineering Dept., Control Systems Lab, Hroon Polytechniou 9, Athens, Greece, 157-80 (corresponding author phone: ++3-210-772-2643; fax: ++3-210-772-1455; e-mail: barthol@central.ntua.gr).

S. L. was with the National Technical University of Athens, Mechanical Engineering Dept., Control Systems Lab. He is now with the GRASP lab at University of Pennsylvania, Walnut Street, Philadelphia, PA 19104-6228 USA.

M.T. Author, is with the University of Karlsruhe, Institute for Process Control and Robotics, 76131 Karlsruhe, Germany.

K. K. Author is with the National Technical university of Athens, Mechanical Engineering Dept., Control Systems Lab.

E. P. Author is with the National Technical University of Athens, Mechanical Engineering Dept., Control Systems Lab.

To address these problems the EU-funded **MiCRoN** project [10] has developed autonomous multi-robot systems that constitute a flexible micro/nanofactory for assembly and manipulation tasks. The **MiCRoN** system is based on a cluster (5-10) of small ($2cm^3$) mobile autonomous robots. These wireless agents, each equipped with onboard electronics, cooperate within a desktop environment to autonomously execute a range of cooperative tasks associated with automated assembly and manipulation in the micro and nano range. A similar concept is presented in the NanoWalker project [11-12], but the motion principle differs considerably and the overall dimensions of the robots are greater. Also, the MINIMAN project [8] refers to a similar concept but the dimension of the robots are much bigger, the robots are tethered and the motion mechanism differs from that of MICRoN.

The main objective of the **MiCRoN** system is the autonomous co-operative operation for micro-robotic tasks. To this aim, a control architecture, which integrates on-line supervision, multi robot navigation, trajectory tracking and position control has been designed, implemented and successfully tested with real hardware. Simulations verify that the control design compensates for relevant micro-scale effects while at the same time achieves the stringent accuracy requirements. Preliminary experimental results on a micro-robotic platform have validated the navigation algorithm properties and the control architecture robustness.

The rest of the paper is organized as follows: Section II provides a brief presentation of the **MiCRoN** robot. Section III describes the control architecture. Section IV presents the controller implementation and section V the kinematic modelling. Section VI presents the simulation and experimental results. Section VII presents the conclusions.

II. The MiCRoN Platform

A. Platform Description

The platform is composed by: (i) A cluster of 5-10 miniature robots. (ii) The power floor, which is responsible for the wireless powering of the robots and which also provides a mechanical interface for the robots. Its dimensions are $20 \times 22cm$ [10]. (iii) The global positioning system, which is based on the Moire-fringes effect and provides accuracy of a few μm [10]. (iv) A central computer, which is responsible for the entire control of the platform and communicates with the robots through IR channel. (v) A graphical user interface that allows the user to program, visualize and control the platform operations.

B. Miniaturization and Micro domain limitations

Several control issues arise due to the miniature size of the robots and due to the peculiar nature of the micro/nano tasks. The most important of these are:

- Power supply constraints due to the miniature-dimensions
- Communication bandwidth limitations that constrain the control and sensor data rate
- Limited on-board processing capabilities
- The inherently limited degrees of freedom of the robot's micro-manipulator
- The limited force capabilities of the robotic mechanism
- The micro-scale effects that might affect micro-manipulation, such as van der Waals, capillary and electrostatic forces [13]

The miniaturization constrains are handled by implementing a centralized control scheme where motion planning and position control are realized off-board by the central PC. The micro-scale effects affecting micro/nano manipulation are treated through the adopted control strategies.

C. Robot design

The robots developed in the project [10], consist of several functional modules as shown on Fig. 1

- 3 DOF (x, y, θ_z) locomotion module
- 1 DOF rotational manipulator
- Tool modules attached as an end-effector to the manipulator
- On-board electronics for driving the sensors and actuators
- Power pack (battery or inductive transmission)
- Targets for the Moire-based positioning system [14]
- IR communication module

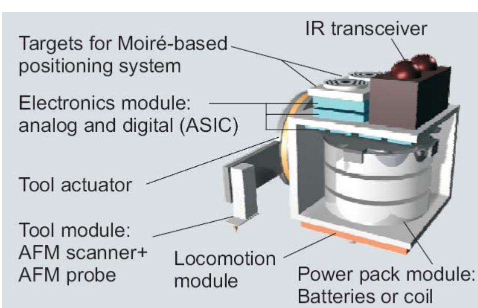

Fig. 1. MiCRoN robot

III. CONTROL SYSTEM ARCHITECTURE

The centralized control architecture implemented is shown in Fig. 2. The user first decomposes the high level tasks into lower level actions. This primitive task sequence is then input to the control system through a Graphical User Interface (G.U.I.), using a task description language specifically developed for the MiCRoN platform (Micron Task Language -MiTL). A lexical parser performs parsing of the input task description and retrieves relevant task data from a task bank to create a sequence of motion and tool commands. The commands are fed to a simulation stage for feasibility verification. Successful task sequences are routed for execution by the system controller. Motion data are wirelessly provided to the micro-robots and a sensor based fusion scheme is then applied to close the control loop.

More specifically, the modules and units that compose the MiCRoN control system architecture, are the following:

1) Graphical User Interface

The Graphical User Interface (G.U.I.) creates an input XML file that contains the input data and which is sent to be processed by the Parser.

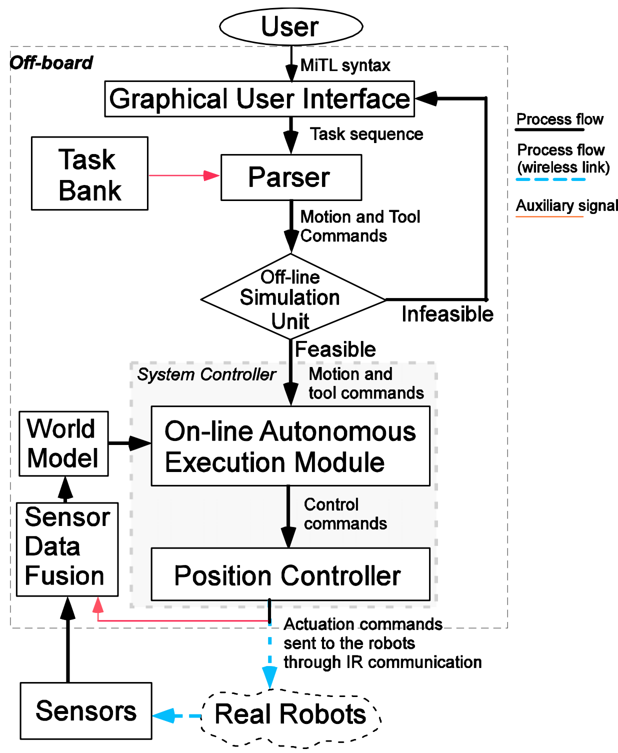

Fig. 2. MiCRoN system architecture flow diagram

2) Parser

This unit, after parsing the XML syntax, generates a data structure, which incorporates motion planning commands, tool commands and operation commands than can be understood by the Simulation Unit and the Autonomous Execution Module. During the parsing process, the input data structure is augmented with complementary data, such as parameter values, stored in the Task Bank.

3) Simulation

This unit performs simulation of the plan. Its purpose is to perform plan feasibility verification prior to autonomous execution and anticipate for (i) workspace and obstacle constraints and (ii) discrepancies between task requirements and available hardware. If the simulation output is successful, the input data structure list is fed to the Autonomous Execution Module for on-line execution; otherwise the user is prompted to reschedule the task sequence.

4) Autonomous Execution Module (AEM)

The AEM is a high-level controller for on-line multi-robot navigation, coordination, cooperation and trajectory tracking. It comprises the following units:

- *Task Supervision Unit:* It supervises the sequential execution of the input task sequence and monitors the execution status of the current process.
- *Motion Planning Unit*: It is responsible for running the algorithms for online multi-robot navigation, coordination and cooperation of the team of micro robots. The output from this unit, is a set of velocities sent directly to the Position Controller.
- *Trajectory Tracking Unit:* It is responsible for performing trajectory tracking of the manipulator's end-effector position. The output is a set of velocities sent directly to the Position Controller.

5) Position Controller

This is the low-level controller, which is responsible for the mobility and manipulability of the robots. It translates the velocity commands into appropriate actuation commands such as voltage signals amplitude and frequency. These are sent through an infrared communication channel to the real robots [10].

6) Sensory module and World Model

The control loop closes with the Sensor Fusion Unit, which acquires sensory data and performs processing and data fusion using Kalman Filter based algorithms in order to provide position estimations to the System Controller through the World Model. (W.M.) The W.M. in turn contains configuration parameters and geometrical data for the robots, the objects and the workspace.

IV. Robot Kinematic Model

The position and orientation of the robot and of its end-effector can be described using the representations of Fig. 3.

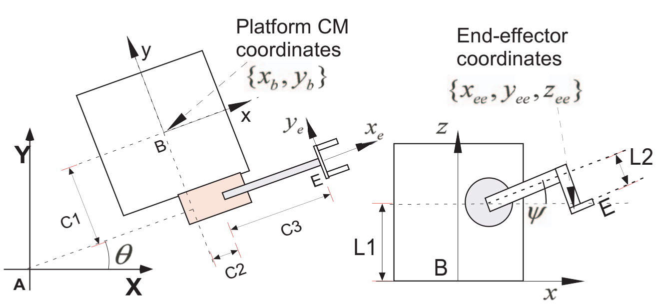

Fig. 3 Platform's top and lateral view

The i^{th} robot's, four degrees of freedom (dof), are expressed, wrt the base frame, using the following set of parameters: $\mathbf{q}_i = \{x_{bi}, y_{bi}, \theta_i, \psi_i\}$. The θ variable represents the angle about the z axis of frame A, and the ψ variable corresponds to the angle of the manipulator wrt the axis x_b of frame B. The configuration of the end-effector wrt the inertial frame, which is attached to the tool center point of the end-effector, is given by $\mathbf{Q}_i^E = \{x_{eei}, y_{eei}, z_{eei}, \theta_i\}$

It is not required to develop a dynamic model of the system, because the platform and actuators dynamic behavior is cancelled out by the machine learning algorithms presented in paragraph V.

V. Controller Implementation

A. On-line multi-robot navigation

The online multi robot controller is realized under the multirobot navigation functions framework [14]. A special controller design that takes into account the micro robot motion principle [15] is applied to increase the micro-robot motion accuracy.

Multirobot navigation functions are a special category of potential functions that have a unique minimum at the destination configuration. Their negated gradient vector field provides for a fast, feedback based closed form solution to the motion planning and the multi-robot motion control problem.

The robot's 4-dof velocities are related to the input signal [14] by the following equation:

$$\dot{\mathbf{q}}_i = \mathbf{R}(\theta_i)\left(\mathbf{u}_i + \mathbf{e}_i(\mathbf{u}_i)\right), \quad \mathbf{u}_i \in D_i \subset \mathrm{R}^3 \qquad (1)$$

where $\mathbf{q}_i = \{x_{bi}, y_{bi}, \theta_i, \psi_i\}$ is the position and orientation vector, $\mathbf{u}_i$ is the input linear and angular velocity, $\mathbf{e}_i(\mathbf{u}_i)$ is the error term, $\mathbf{R}(\bullet)$ is a rotation matrix, and D_i is the favored regions set for the i'th robot. We use

$$\varphi = \varphi(\mathbf{q}\ , \mathbf{q}_d) \qquad (2)$$

as a multirobot navigation function, where $\mathbf{q}\ = [\mathbf{q}_1^T ... \mathbf{q}_n^T]^T$ is the augmented vector of the current robot configurations, $\mathbf{q}_d$ is the augmented destination configuration vector of the robot team, and n is the number of robots. The implemented control law is of the form (see [14] for details):

$$\mathbf{u}_i = f\left(-\nabla\varphi, \mathbf{q}_{i,} D_i\right) \qquad (3)$$

This controller produces actuation signals in the favored velocity region of each robot according to the vector field produced by the negated gradient of the multirobot navigation function. It features theoretically established properties of global convergence and collision avoidance. The user only needs to define the desired destination configuration of the multirobot system. The coordination and cooperation between the robots is dynamically produced by the navigation vector field depending on the robot proximity relations. By exploiting the D_i set of each agent, the controller keeps the actuation error low, and results in faster convergence and increased positioning accuracy. An example of such a D_i set is shown in Fig. 6 as the innermost dark-grey regions of the actuation space.

Due to the feedback based character of the utilized control law, the methodology is very robust to sensor noise and model uncertainties, a feature that is very useful for the microrobotic platform in overcoming the micro scale effects.

B. Trajectory Tracking

There are several manipulation tasks, which require the end effector to move from an initial posture to a final one, in a specified time interval, subjected to acceleration and velocity constraints. This can be accomplished via trajectory planning of the end-effector, using blended polynomial time laws [16]. This method generates smooth trajectories and directly verifies whether the resulting velocities and accelerations of all degrees of freedom can be supported by the micro-robot. Fig. 4 depicts the block diagram for a single robot trajectory tracking.

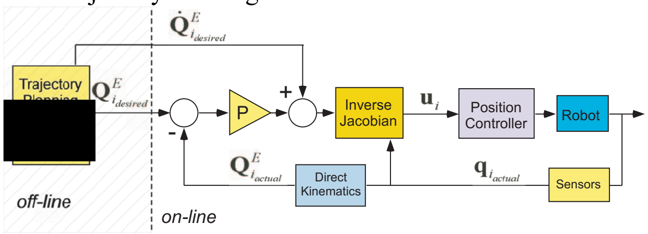

Fig. 4. Trajectory tracking control loop

Analytical expressions of the end-effector trajectories are derived off-line in the cartesian space. During execution the estimated end-effector position error is processed by a P controller and added to a feed-forward term. The trajectory tracking controller is of the form:

$$\mathbf{u}_i = J_i^{-1}\left(Q_i^E\right)\left(\dot{Q}_{i_{des}}^E(t) + \mathbf{P}\left(Q_{i_{des}}^E(t) - Q_i^E\right)\right) \quad (4)$$

where $Q_{i_{des}}^E(t)$ is the given trajectory for the i^{th} robot end-effector, J_i the 4×4 Jacobian matrix relating $\dot{Q}_i^E(t)$ to $\dot{q}_i(t)$, and P a positive definite matrix.

This tracking controller does not possess obstacle avoidance properties, since the considered tracking tasks are performed locally in an obstacle free neighborhood. Although we could have derived a tracking controller exploiting the whole D_i region of every robot, we opted to only use a compact, polygonal shaped subset $\Pi_i \subset D_i$ of the D_i set centered at the origin. This was needed to overcome the communication delays that would have otherwise allowed for significant deviations of the end-effector from the specified trajectory, causing undesirable side effects (like tool bending or breaking or specimen dislocation).

This tracking controller gives exponential tracking capabilities to the end effector, under the assumption that the trajectory velocities satisfy: $\dot{\mathbf{q}}_{i_{des}}(t) \in \Pi_i$ and that the noise in the sensors is substantially suppressed.

C. Position Control

The position controller is a low-level component of the control system. It translates the high-level velocity commands (v_x, v_y, v_θ) of the robot platform as well as the dof of the manipulator into actuation commands. The design of the position controller is characterized by its flexibility:

- The controller can interact with robots of different numbers of DOF
- Different motion behaviors can be dealt with
- Nonlinearities and singularities are being compensated

In order to avoid unstable control conditions, two closed loop control schemes are not suitable. Instead, the low level control runs in open loop mode. The design of the position control then consists in finding a control law that maps velocity commands into actuation commands:

$$\mathbf{r}_j = \sum_i f_i(\mathbf{u}_i) \quad (5)$$

where $\mathbf{r}$ is an actuation command, j corresponds to the number of actuation commands and i to the number of velocity commands. The actuation commands relevant for the MiCRoN robots are the voltage and frequency of each actuator.

The position controller can be taken as a mathematical model of the inverse actuation. Although there are plenty of approaches to find a mathematical model, the most promising are in the domain of artificial intelligence and machine learning. As the controller is open-loop, the calibration of the model is done offline. The data is collected by measuring the relation

$$\mathbf{x}_i = \sum_j f_j(\mathbf{r}_j) \quad (6)$$

where $\mathbf{x}_i$ is the resulting velocity of the robot when applying the actuation command $\mathbf{r}_j$. A position measurement system based on Moiré marks [17] has been used to collect the velocity data. It provides position information with a precision of a few µm.

Figure 6 depicts the influence of one of the actuator commands on the translational motion of the platform of a MiCRoN prototype. For controlled planar motion more than one actuator should be simultaneously activated, which suggests that for measurement purposes all possible combinations of actuator signals have to be investigated.

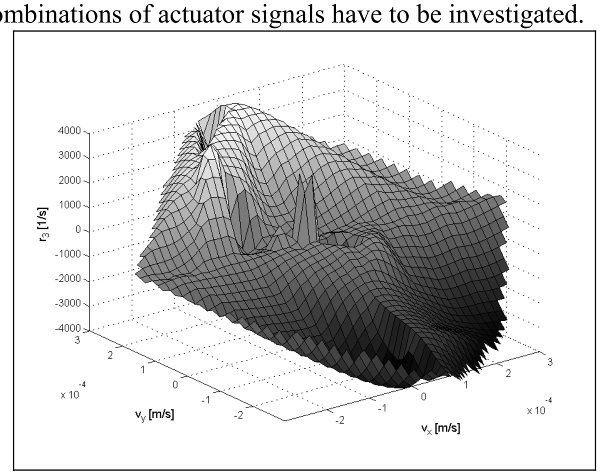

Fig.5: The frequency of one of the robots actuators with respect to v_x and v_y at constant voltage. All other actuators are deactivated (no signal)

The data set obtained has been divided into a training and a validation set at which cross validation has been performed. The following results refer to the validation set.

For the building of the model, an approach based on genetic programming has turned out to be the most successful. Genetic programming mimics processes observed in nature in order to improve the fitness of an individual [18][19]. In contrast to neural network algorithms where only parameters are optimized, in genetic programming the individual is a model as a total (its equations and parameters). Hence, no prior knowledge about the nature of the model is required. Moreover, nonlinearities and discontinuities can be modeled easily.

Figure 6 shows the error distribution obtained by a genetic program. The relative error is extremely low at low velocities. This is a desirable characteristic because this is the region of high precision motion. For higher velocities, the model error rises up to 23% of the maximum velocity.

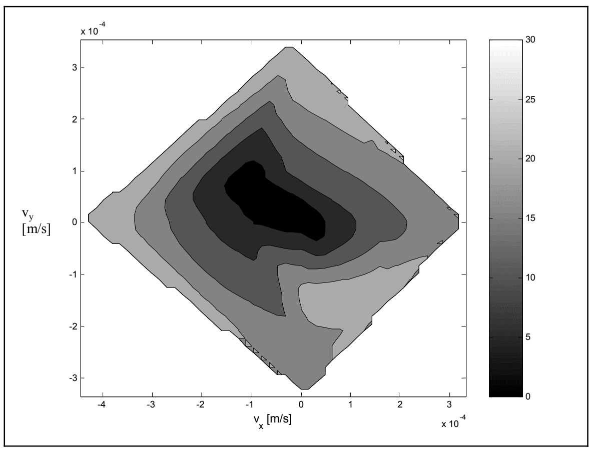

Fig. 6: Relative overall error [in %] of position controller with respect to v_x and v_y

Additional tests with neural networks in place of genetic programs have produced comparable results with a slightly inferior performance in the low velocity domain.

VI. Simulations and Experiments

Theoretical results for the on-line autonomous execution, the trajectory tracking controller and the position controller have been validated also through extensive simulations. The performance of the integrated high level and low level control system has been assessed through extensive simulations that incorporated all the modules included in the off-board control system architecture (see Fig 2).

The control system cannot be experimentally validated on the **MiCRoN** platform, because the robots hardware integration has not reached a fully operational state yet. Nonetheless, preliminary validation experiments have been conducted on the MINIMAN micro-robotic platform [8]. The two micro-robotic platforms bear many differences, but they both employ the same motion principle i.e. stick and slip, and more important both systems perform micrometer precision motion. Taking into account these similarities, preliminary experiments have been conducted to assess the entire control system integration and further to validate the control system capabilities for:

- navigation and
- collision avoidance

Simulation and experimental results

The experiment comprises a single navigation task, and involves two robots: the MINIMAN-3 and MINIMAN-4 which are depicted in Fig 7. The workspace has rectangle geometry ($x = 200mm, y = 300mm$) and the origin of its frame of reference is located at its geometrical center.

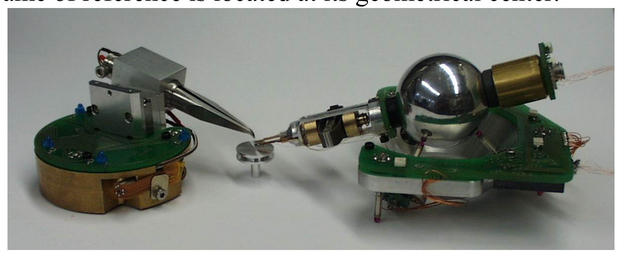

Fig 7. Miniman 4 and Miniman 3

This task requests from each robot to diagonally transverse the area of the workspace on which they lie and switch places as described by the following table:

Table I: Robots Initial conditions and goals

	MINIMAN 3			MINIMAN 4		
	x mm	y mm	θ rad	x mm	y mm	θ rad
Initial Cond.	29.5	43.6	0.175	-22.4	-62	-1.5
Goal	-90	-40	0.175	100	50	-1.6

Also, a static circular obstacle of diameter 10mm was located at point $x = -10mm, y = 0$ on the workspace, in order to demonstrate in an obvious manner the collision avoidance properties of the navigation algorithm.

This task is first simulated in order to verify its feasibility. Immediately after the verification, the task input specifications are automatically sent to the on-line autonomous execution unit and the on-line execution process commences.

The simulation results are presented in the plots of Fig. 8. The first plot depicts the x-y path of the two robots. The second plot presents the trajectory of the angle. Simulation clearly demonstrates the convergence and collision avoidance properties of the high level controller. Fig. 9, presents the experimental results for the same task. As in the case of simulation, the micro-robots converge and perform collision avoidance. The discrepancy of the paths between the simulation run and the execution run for the same initial conditions, is naturally due to the fact that there are uncertainties of the actual system (mostly caused by micro-scale effects and by instabilities in the sensory module) that are not modelled in the motion model and sensor model. The proposed closed loop feedback based, control architecture exhibits robustness characteristics and is able to accommodate those uncertainties and drive the system to the destination configuration. It should be noted that a significant outcome of the experiments is the demonstration of the performance of (i) the navigation and position controllers, (ii) the sensory module and (iii) the world model, all integrated into a well-structured, centralized, robust control architecture, resulting in a stable behavior.

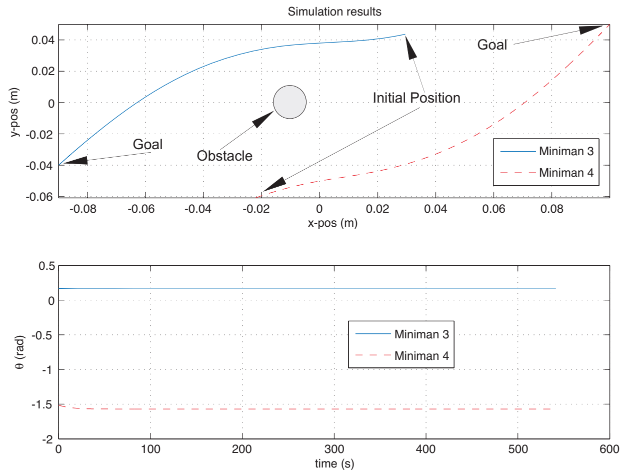

Fig. 8 Simulation results

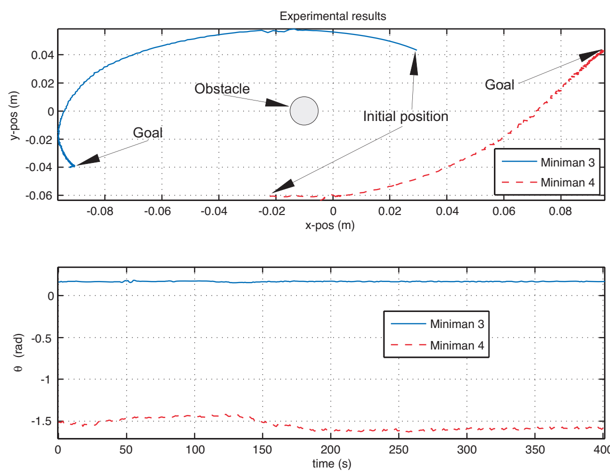

Fig. 9 Experimental results for the same initial conditions as for the simulation case

VII. Conlusions and Future Work

This paper presented the control system architecture for the multi agent micro-robotic platform **MiCRoN**. The entire control system integrates sensory modules, modeling modules, and controller modules. The latter perform decomposition of high level tasks into low level actions which are further processed by the high-level controller responsible for navigation, coordination and cooperation and by the low-level controller, which performs position control exploiting machine learning algorithms. The control system handles miniaturization constrains by implementing hierarchical centralized control architecture. Furthermore, micro-scale effects were treated by features of the navigation algorithm. The controller has also trajectory tracking capabilities for obstacle free neighborhoods. Through experiments and simulations the performance of the complete control system was validated.

As soon as the hardware integration of the **MiCRoN** robots is completed, several navigation and manipulation experiments shall be conducted on the **MiCRoN** platform.

References

[1] Y. Sun and B.J. Nelson, “Biological Cell Injection Using an Autonomous Microrobotic System,” The International Journal of Robotics Research , Vol 21, pp861-868, 2002.

[2] E.A. Branchaud, J. Cham, et al, “A Miniature Robot for Autonomous Single Neuron recordings,” The international Conference of Robotics and Automation (ICRA), 1932-1938, Barcelona, April 2005

[3] H. Chen, Xi Ning, Li Guangyong, J. Zhang and M. Prokos, “Planning and Control for Automated Nanorobotic Assembly,” The International Conference of Robotics And Automation (ICRA), pp 170-175, Barcelona, April 2005.

[4] Li Guangyong, Xi Ning, H. D. Wang, “Functionalized nano-Robot End Effector for in situ Sensing and Manipulation of Biological Speciment,” The International Conference of Robotics And Automation (ICRA), pp 450-455, Barcelona, April 2005.

[5] J.Z. Jiangbo, Li Guangyong, “Modeling and Control of Active End Effector for the AFM Based Nano Manipulators,” The International Conference of Robotics And Automation (ICRA), pp 164-169, Barcelona, April 2005.

[6] K. Deok-Ho, Y. Seok and K. Byungkyu, “Mechanical Force Response of Single Living Cells using a Microrobotic System,” The International Conference of Robotics And Automation (ICRA), pp 5013-5018, New Orleans, April 2004.

[7] Jean-Marc Breguet, Reymond Clavel, Stick and Slip Actuators: design, control, performances and applications. International Symposium on Micromechatronics and Human Science (MHS), (Nagoya), 1998, 89-95.

[8] F. Schmoeckel, S. Fatikow, Smart Flexible Microrobots for Scanning Electron Microscope (SEM) Applications. SPIE’s 7th Int. Symp. On Smart Structures and Materials: Integrated systems, Newport Beach, California, USA, 5-9 March 2000.

[9] H. Aoyama, O. Fuchiwaki, “Flexible Micro-Processing by Multiple Micro Robots is SEM,” The International Conference of Robotics And Automation (ICRA), pp 3429-3435, Seoul Korea, April 2005.

[10] J. Brufau, M. Puig-Vidal, J.Lopez-Sanchez et al., “MICRON: Small Autonomous Robot for Cell Manipulation Applications,” The International Conference of Robotics And Automation (ICRA), pp 856-861, Barcelona, April 2005.

[11] S. Martel et al., “Three-legged Wireless Miniature Robots for Mass-scale Operations at the Sub-atomic Scale,” Proc. of the 2001 IEEE International Conference on Robotics & Automation, Seoul, Korea, May 21-26, 2001.

[12] S. Martel , “Fundamental Principles and Issues of High-speed Piezoactuated Three-legged Motion for Miniature Robots Designed for Nano-scale Operation,” The International Journal of Robotics Research, Vol, 24, No 7, pp 575-588, July 2005.

[13] A. Menciassi, A. Eisinberg, I. Izzo, and P. Dario, “Automatic Microassembly System Assisted by Vision Servoing and Virtual Reality,” IEEE/ASME Transactions on Mechatronics, Vol. 9, No. 2, pp 311-320, June 2004.

[14] S.G. Loizou and K.J. Kyriakopoulos “Closed Loop Navigation for Multiple Holonomic Vehicles”, IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 2861-2866, 2002

[15] S. G. Loizou K. J. Kyriakopoulos, “Navigation of Multiple Input Constraint Micro-Robotic Agents”, IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 1670-1675, 2005

[16] L. Sciavicco, B. Siciliano, “Modelling and Control of Robot Manipulators”,Springer, 2nd edition, 2002.

[17] R. Estaña, H. Woern, “Moiré-Based Positioning System for Micro Robots,” SPIE’s Int. Conference on Optical measurement Systems for Industrial Inspection III (2003), Munich, Germany, S. 431-442

[18] J.R., Koza: "Genetic Programming: On the programming of computers by means of natural selection.", MIT Press, 1992.

[19] G. Gray, D. Murray-Smith,; K. Sharman, Y. Li,"Nonlinear model structure identification using genetic programming" In: Late breaking papers of Genetic Programming '96. Stanford CA, 1996.