Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, August 12-17, 2007

A Neural Network-based Learning Controller for Micro-sized Object Micromanipulation

Mohsen Shahini, William W. Melek, John T. W. Yeow

***Abstract*— In this paper, automated micro-sized objects manipulation is investigated. The novelty of the proposed method lies on the compensation of all the nonlinear scaling forces which are dominant over gravitational force. A dynamic neural network has been added to a PD conventional controller for automated micromanipulation control. Weight-updating rules have been obtained in such a way that the system is uniformly ultimately bounded (UUB) in the sense of Lyapunov. Simulation results for controlled pushing of a micro-object have been illustrated and the efficiency of the method has been shown by comparing its result with that of a linear controller.**

I. INTRODUCTION

Miniaturization of systems and tools has been much realized in recent decades. The main drive toward micro/nano electromechanical systems (MEMS/NEMS) has been the ever-increasing need to save energy and material which can be best achieved by almost weightless miniaturized devices. They also help making special instruments that execute tasks which are otherwise impossible to realize. These advantages, however, have been accompanied by some big challenges: Gravity and viscosity in Newtonian's framework which have served best in prediction of macro-scaled objects trajectory now are beaten by other less-known surface forces such as van der Waals, Kondo effect, capillary adhesion, electrostatic, etc. As of today, there has not been any description which could even roughly foresee the continuous states (position and velocity, for instance) of particles and micro objects under manipulation.

High precision positioning of micro and nano particles is highly demanded of micro-assembly [1], cell and DNA manipulation [2] and molecular interaction studies[3]. A lot more fascinating capabilities of high precision micro robotics in biomedical applications, such as drug delivery, cell healing[4], and other surgical processes inside body has become under attention for some years[5], [6]. For most of these potential applications to materialize, characterizing the dynamics of molecular forces is inevitable. Despite some efforts to capture the mathematical model of these seemingly week surface forces, much remains to be done before one could base a micromanipulation controller on any analytical model. That explains why most of the micromanipulation workstations either require a careful surveillance by an operator [1], or are Tele-operated [7]. For most of the existing setups, different trials have to be carried out for the cells and particles to be positioned accurately.

There are countless types of mili and submili-sized insects who can very dexterously manipulate micro objects. This fact implies that they must somehow know the information on micro forces which we do not. Although part of this information may come genetically with an insect when it is born, research on Drosophila has revealed that the relatively simple structure of most insects' brain has the capability of learning[8]. Given the relative simplicity of the structure of a bug's brain, we are highly inspired to apply artificial neural networks as an adaptive control tool to capture the behavior of poorly-known surface forces. Utility of neural networks as an intelligent method in compensation of unknown, nonlinear friction force in conventional robot manipulation has been represented in literature [9]. Obviously, friction in macro systems usually counts only for a small proportion of the dynamic system which in a lot of cases treated as a disturbance or noises in the controller loop. In contrast, micromanipulation deals with a system whose response almost entirely stems from the non-inertia forces. As such, application of artificial neural networks in such a system is a novel work which requires additional systematic investigation.

This paper has been organized as follows: In section II we introduce the problem of "pushing" as one of the fundamental way of micro-object manipulation. In section III is proposed a controller strategy which employs artificial neural network for micro particle pushing. Boundedness of its state variables is discussed and a Lyapunov function is defined to mathematically guarantee boundedness of displacement error in pushing. In section IV, results from numerical analysis are shown and comparison is made against a PD conventional controller. This paper concludes with some remarks and future works in section V.

II. BACKGROUND

Finding the physical variables of a system as input-output relationship is the crucial step to analyze its behavior. Basically, pushing the block of Fig.1 can be formulated simply as:

$$m\ddot{q} + F_f + F_d = F \tag{1}$$

where F, F_f, F_d, m and $\ddot{q}$ are delivered force by microactuator, resistance force in contact area, disturbance, mass and

Mohsen Shahini is PhD candidate at the University of Waterloo, Department of Mechanical Engineering, N2L 3G1, Canada, phone: 519 888-4567 Ext 37577; fax: 519 888-4333; e-mail: mshahini@engmail.uwaterloo.ca,

William W. Melek is Assistant Professor at the University of Waterloo, Department of Mechanical Engineering, N2L 3G1, Canada, e-mail: wmelek@mecheng1.uwaterloo.ca

John T.W. Yeow is Assistant Professor with the Department of System Design Engineering, University of Waterloo, N2L 3G1, Canada, e-mail: jyeow@engmail.uwaterloo.ca

resulting acceleration, respectively. The term, F_f , which is also referred to as surface scaling force, is complicatedly affected by parameters such as surface roughness, temperature, humidity, velocity, applied force, geometry of surface, materials, electrostatic charge, etc. That makes it impossible to derive a model for this surface force. On the other hand, unlike conventional robotics, this force is considerably large in comparison to the well-known gravitational force resulting from the mass so that it can not be treated as a negligible term, or at most, added to the disturbance on the system. It is hence advantageous to define a learning mechanism which can over some experimental trials be trained to fairly estimate this force under given operating condition.

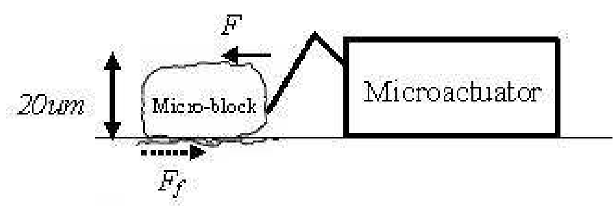

Fig. 1. Schematic of micro-object pushing

In[10], we introduced a comprehensive investigation to bring together all influential parameters necessary to determine the overall micro surface force. This force can be represented as a function of the following variables:

$$F_f = f(\dot{q}, \overline{R_a}, E, \nu, RH, T, A, F_a, m) \quad (2)$$

where $\dot{q}$ is relative velocity of the two surfaces in contact, $\overline{R_a}$ represents surface roughness, RH is relative humidity, T is temperature, E, ν, A and m are Young's modulus, Poisson ratio, surface area in contact and mass of the micro object, respectively. F_a is the applied force. It should be noted that physical parameters which are unlikely to vary during pushing has omitted as variables.

III. Control Strategy

A. NN Controller Design

In order to design the stable NN controller, the error dynamics of the manipulator ought to be first considered. In adaptive control of a manipulator, an auxiliary filtered tracking error signal is often required to remove the acceleration components from the dynamic equations. With the same idea Lewis [11] introduced a filtered error to the NN control. Here, a simple stable NN controller with the filtered error is proposed to control a micromanipulator based on Lyapunov function, which can guarantee the uniform ultimate boundedness (UUB) of the closed-loop system under some assumption. The convergence of the tracking error is also guaranteed.

Given the desired tip-position trajectory, the tracking error and its derivative are:

$$e = q_d - q$$

$$\dot{e} = \dot{q}_d - \dot{q}$$

We define the filtered tracking error as:

$$r = \dot{e} + ke \quad (3)$$

with $k > 0$. Differentiating (3) and invoking (1), it is seen that the pushing dynamics can be described in terms of the filtered tracking error as:

$$m\dot{r} = kmr + F_f + F_d - F + m\ddot{q}_d - k^2 me \quad (4)$$

To set a NN-based controller, let:

$$F = \hat{F}_f + k_v r + m\ddot{q}_d - k^2 me \quad (5)$$

where $\hat{F}_f$ is an estimate of F_f , and $k_v r = k_v \dot{e} + k_v ke$ an outer PD tracking loop.

It should be noted that one could add another robust control term to (5) to robustify the controller against unmodeled disturbance [9].

Fig. 2 is a block diagram of the mechanism of compensation for the unmodeled scaling forces. The nine-input single-output neural network will learn online by adjusting the weights to compensate the effect of scaling forces. The filtered tracking error is aggregated with the estimated friction to generate an input force to the manipulator. Since the effect of the mass of micro-sized objects is completely suppressed by tracking error and estimated scaling forces, the term $m\ddot{q}_d - k^2 me$ is safely negligible and can be omitted.

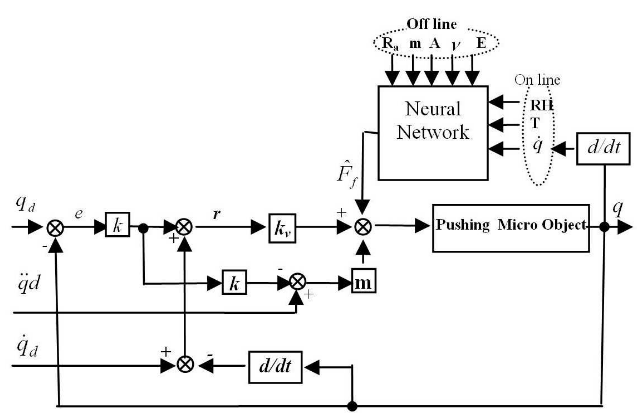

Fig. 2. Schematic block diagram of the pushing control with a NN-based friction compensator

B. NN Approximation

According to the universal approximation property of neural networks [13], there exist a two-layer NN such that:

$$F_f = W^T \sigma(V^T x) + \varepsilon \quad (6)$$

where the weight matrices W and V are unknown to be tuned in an unsupervised manner during pushing. These ideal target weight matrices are not necessarily unique. The approximation error is bounded on a compact set by $\varepsilon \leq \varepsilon_N$, with ε_N a known bound. The activation function σ at

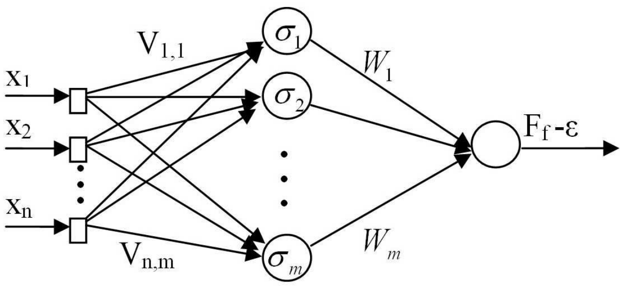

Fig. 3. Structure of the proposed two-layer Neural Network

the hidden layer is a nonlinear function (typically sigmoid function).

Since the size of neural network is difficult to determine, often the weight estimates of $\hat{W}$ and $\hat{V}$ are used with a certain size to approximate the function. That is:

$$\hat{F}_f = \hat{W}^T \sigma(\hat{V}^T x) \tag{7}$$

Then, the following weight estimation errors can be defined as:

$$\tilde{W} = W - \hat{W}, \tilde{V} = V - \hat{V} \tag{8}$$

The functional approximation errors can be defined as:

$$\tilde{F}_f = F_f - \hat{F}_f \tag{9}$$

And the hidden layer output error is described as:

$$\tilde{\sigma} = \sigma - \hat{\sigma} \tag{10}$$

where $\sigma = \sigma(V^T x)$ and $\hat{\sigma} = \sigma(\hat{V}^T x)$ and its Taylor series expansion about $\hat{V}^T x$ is:

$$\sigma = \hat{\sigma} + \sigma'(\hat{V}^T x).(\tilde{V}^T x)^2$$
$$\Rightarrow \tilde{\sigma} = \sigma'(\hat{V}^T x).(\tilde{V}^T x) + O(\tilde{V}^T x)^2 \tag{11}$$

C. Training Algorithm

To prove the stability of the control system which is augmented by a NN, several assumption and conditions have to be considered. Those are outlined in this section.

Assumption 1 (Bounded Reference Trajectory): The desired pushing trajectory is bounded so that:

$$\left\| \begin{array}{c} q_d(t) \\ \dot{q}_d(t) \\ \ddot{q}_d(t) \end{array} \right\| \leq q_B \tag{12}$$

with q_d , $\dot{q}_d$, $\ddot{q}_d$ the desired trajectory, desired velocity and desired acceleration, respectively, and q_B a known scalar bound.

Lemma 1 (Bound on NN input x): For each time t, the NN input vector $x(t)$ is bounded.

Proof: the NN input vector x is:

$$x = [\dot{q} \quad R \quad E \quad \nu \quad RH \quad T \quad A \quad m]^T$$
$$= [(\dot{q}_d - r + ke) \ \ R \ \ E \ \ \nu \ \ RH \ \ T \ \ A \ \ m]^T \tag{13}$$

During pushing each micro particle, R, E, ν, A, and m remain unchanged ($= c_1 > 0$). Equivalently from the assumption 1, one can extract $\|\dot{q}_d(t)\| \leq \dot{q}_B$. Temperature and relative humidity are evidently bounded by some constant positive value c_2 in total. Therefore:

$$|x(t)| \leq c_1 + c_2 + \dot{q}_B + k\,|e| + |r| \tag{14}$$

The solution of the system (3) with the initial value $e(t_0) = e_0$ is:

$$e(t) = e_0 \ \exp^{-k(t-t_0)} + \int_{t_0}^{t} \exp^{-k(t-\tau)} r(\tau)\, d\tau \tag{15}$$

Thus,

$$|e| \leq |e_0| + \frac{|r|}{k} \tag{16}$$

Then the bound can be derived as:

$$|x(t)| \leq c_1 + c_2 + \dot{q}_B + k\,|e_0| + 2\,|r| \tag{17}$$

Then the NN input vector is bounded as long as the controller guarantees that the filtered error $r(t)$ is bounded.

Theorem 1: Consider the dynamic system of micro manipulator described by (4), for the bounded, continuous desired tip trajectory with bounded velocity and acceleration. NN controller (5) can guarantee the uniform ultimate bundedness of the close-loop system with the gains satisfying $k_v - km > 0$, and the NN weight tuning algorithms given by:

$$\begin{array}{l} \dot{W} = F\hat{\sigma}r \\ \dot{V} = Gx(\hat{\sigma}'^T \hat{W} r)^T \end{array} \tag{18}$$

where F, G are the constant positive definite matrices. Moreover, the weight estimates $\hat{W}$ and $\hat{V}$ are bounded and the filtered tracking error r goes to zero asymptotically.

Proof: By substitution of (7) to (5), the control signal would become:

$$F = \hat{W}^T \sigma(\hat{V}^T x) + k_v r + m\ddot{q}_d - k^2 me \tag{19}$$

Now, by substitution of (19) and (6) into the error dynamics, i.e. (4) and using the estimation errors in (8),(9), and (10), one obtains:

$$m\dot{r} = (km - k_v)r + W^T\sigma - \hat{W}^T\hat{\sigma} + (\varepsilon + F_d) \tag{20}$$

Adding and subtracting $W^T\hat{\sigma}$ yields:

$$m\dot{r} = (km - k_v)r + \tilde{W}^T\hat{\sigma} - W^T\tilde{\sigma} + (\varepsilon + F_d) \tag{21}$$

Adding and subtracting $\hat{W}^T\tilde{\sigma}$ in (21) yields:

$$m\dot{r} = (km - k_v)r + \tilde{W}^T\hat{\sigma} + \hat{W}^T\tilde{\sigma} + \tilde{W}^T\tilde{\sigma} + (\varepsilon + F_d)$$

Substituting from (11) gives:

$$m\dot{r} = (km - k_v)r + \tilde{W}^T\hat{\sigma} + \hat{W}^T\hat{\sigma}'\tilde{V}^T x + \omega(t) \tag{22}$$

where

$$\omega(t) = \tilde{W}^T\hat{\sigma}'\tilde{V}^T x + W^T O(\tilde{V}^T x)^2 + \varepsilon + F_d$$

is called the disturbance term. One should note that the pushing disturbance F_d, NN approximation error ε, and the higher order terms in Taylor series expansion all have the same weights as disturbance in the error system.

Several conditions must be imposed on the error system in (22):

Condition-1: k should be positive definite.

Condition-2: Weights (W and V) and activation function σ should be bounded.

Condition-3: The first derivative of activation functions should be bounded as well.

To satisfy the last two conditions, sigmoid function is a good candidate since this function and its first differentiation are both bounded by 1. Let the Lyapunov candidate to be defined as:

$$L = \frac{1}{2}mr^2 + \frac{1}{2}tr(\tilde{W}^T F^{-1}\tilde{W}) + \frac{1}{2}tr(\tilde{V}^T G^{-1}\tilde{V}) \quad (23)$$

Differentiating and substitution from (22) will yield:

$$\dot{L} = r[(km - k_v)r + \tilde{W}^T\hat{\sigma} + \hat{W}^T\hat{\sigma}'\tilde{V}^T x] + tr(\tilde{W}^T F^{-1}\dot{\tilde{W}}) + tr(\tilde{V}^T G^{-1}\dot{\tilde{V}})$$

For the sake of simplicity, we have assumed that the influence of disturbance $\omega(t)$ is negligible. Since,

$$r\tilde{W}^T\hat{\sigma} = tr(\tilde{W}^T\hat{\sigma}r)$$
$$r\hat{W}^T\hat{\sigma}'\tilde{V}^T x = tr(\tilde{V}^T x r\hat{W}^T\hat{\sigma}')$$

then, one can write:

$$\begin{aligned}\dot{L} \quad = \quad & -(k_v - km)r^2 \\ & + tr\{\tilde{W}^T(F^{-1}\dot{\tilde{W}} + \hat{\sigma}r)\} \\ & + tr\{\tilde{V}^T(G^{-1}\dot{\tilde{V}} + xr\hat{W}\hat{\sigma}')\} \qquad (24)\end{aligned}$$

If tuning algorithms (18) hold, the last two terms in the right hand side of (24) would become zero. Note that, since W and V are constants, $\dot{\tilde{W}} = -\dot{\hat{W}}$ and $\dot{\tilde{V}} = -\dot{\hat{V}}$. Then:

$$\dot{L} = -(k_v - km)r^2$$

The parameter $(k_v - km)$ is always positive as long as gains of the outer PD tracking loop satisfy $k_v > km$. This leads to $\dot{L} \leq 0$, and since $L > 0$, one can deduce the stability in the sense of Lyapunov (SISL). Moreover, considering (16) one can conclude the boundedness of all the terms in right hand side of (4) and consequently the boundedness of $\dot{r}$. As such, $\ddot{L} = -2(k_v - km)r\dot{r}$ is bounded and thus, $\dot{L}(t)$ is uniformly continuous. Applying Barbalat's Lemma [9] we can conclude that $\dot{L}$ should essentially approach to zero as t goes to infinity and as a result, $r(t)$ vanishes.

IV. Simulation Results

To validate the proposed neural network control strategy, in this section simulation of a micro-object manipulation is presented. The nonlinear equation for F_f was employed from a friction model proposed by [14] with some slight changes to the modeled constants. Pushing of a micro-sized cube aluminum with the dimension of $65\mu m \times 65\mu m \times 100\mu m$ along a sinusoidal desired trajectory for a distance of $100\mu m$ was simulated with a PD controller with and without neural network compensation loop as showed in Fig. 2. PD gains were tuned first for a PD-only controller to give the minimum mean square error (MSE) over a specific range of gains. Then gains were kept unchanged. A two-layer NN with 8 hidden nodes was first trained and tested using hill-climbing back propagation learning. Training sets were vectors of friction forces in different velocities and constant temperature at 25 Celsius and constant relative humidity at 75%, calculated from the model proposed in [14]. Other input parameters in (2) were assumed unchanged. Then the NN model was embedded into the control loop in Fig. 2.

Positioning error of micro object pushing along the trajectory shown in Fig. 4 are depicted in Fig. 5 for conventional PD and off-line trained neural network. As seen, with ambient condition kept unchanged, neural network has almost completely cancelled out the error from the nonlinear part of the dynamic equation. This figure indicates the superiority of NN-based compensation over linear PD controllers.

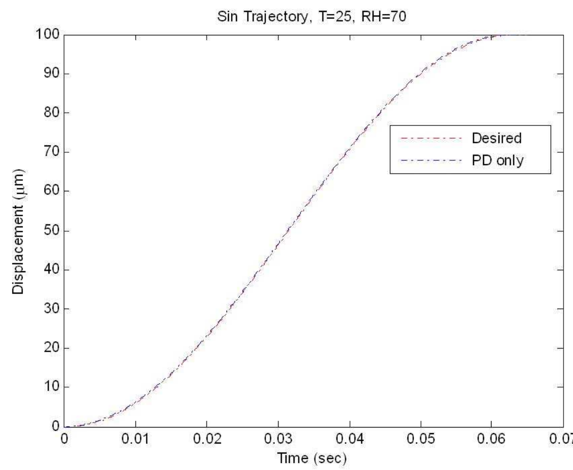

Fig. 4. Micro-sized object pushing trajectory

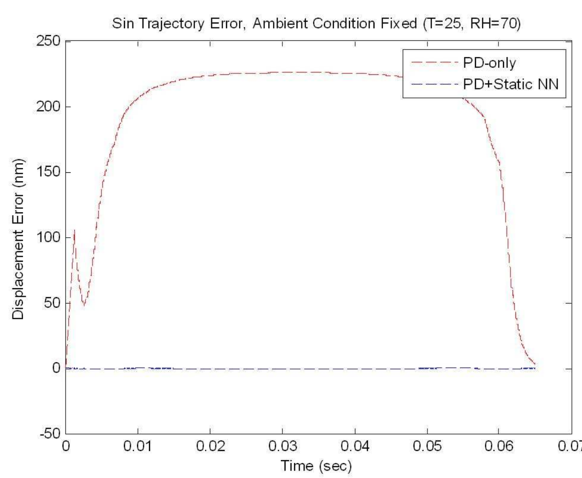

Fig. 5. Positioning error for controlled pushing under the same ambient condition as data collection for off-line NN training

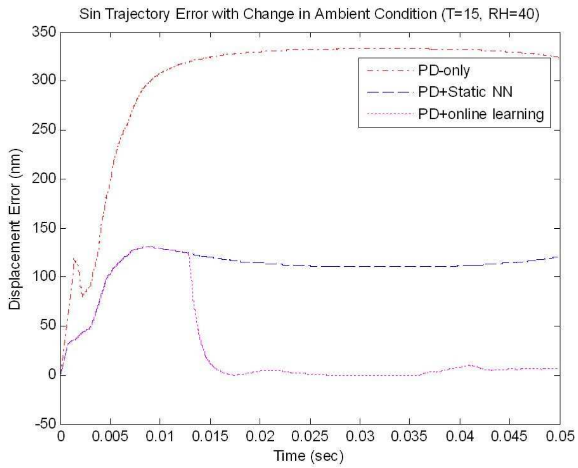

Fig. 6. Positioning error for controlled pushing in different ambient condition from data collection

Since scaling forces are highly subject to big changes with small variation in ambient conditions or other unknown parameters, we propose to tune the weights of off-line trained neural network in real time with the algorithm discussed in section III. The same trajectory simulation was repeated with the previous neural network, but temperature and relative humidity fell down from 25 Celsius degree and 70% to 15 Celsius degree and 40% respectively. This resulted in about 10 to 20 percent deviation in the modeled scaling forces in different velocities. Results of the displacement error are illustrated in Fig. 6. While the error profile of the conventional PD controller shows the same trend as in Fig. 5, the error of static neural network suggest that it has not been trained to compensate the dynamic scaling force in the new ambient condition, and therefore dramatically loses its capability of canceling out the nonlinear error. This necessitates an unsupervised learning of the proposed NN in a real-time fashion. Thus, online learning was implemented on outer-layer weight matrix of W by learning rate of $F = 0.05$ and input layer weight matrix of V was kept unchanged. W was not tuned at two small periods in the inception and the end of pushing when the velocity was not high enough. Derivative of the error is oscillatory and undesirably large in low velocities and feeds back a big signal to the weight tuning loop. This was because of the difficulty in modeling the static friction which in reality exists at ”zero” velocity but in modeling it exists when velocity is less than a ”threshold”. Stick-slip is a common term to refer to this phenomenon at very low speed of pushing [15]. It can be clearly seen in Fig. 6 that learning has begun around $t = 0.013sec$ after which neural network has rapidly adapted its weights to let the controller achieve accurate positioning of the micro object.

V. CONCLUSION

In this work, we tackled the problem of controlling micro-sized object pushing as the first stride toward automated micro-sized object manipulation. Neural network as an intelligent tool was used in the absence of any reliable models for scaling forces. All parameters affecting these forces in micro-domain were studied and considered as potential to the input vector of the neural network block. In addition, online learning rules for weight matrices of a two-layer neural network were derived to assure that the closed-loop system is stable in the sense of Lyapunov. A complex nonlinear model was adapted to represent the micro forces, as a function of velocity, temperature and humidity. Simulation results confirmed the efficiency of scaling force compensation by an adaptive neural network. Some problem of inefficient online learning was observed at low velocities where the modeled scaling forces are acting as a piece-wise function of velocity. To solve this problem, some preliminary studies on capability of a neural network with modular structure in scaling force approximation have been carried out. We believe a good structure of a modular neural network is a better candidate for the learning controller, especially at zero and low velocities regions where discontinuity is present. In the continuation of the current research, implementing the proposed controller in a MEMS-micro actuator to precisely push a micro object in an experimental setting is being currently investigated.

REFERENCES

[1] N. Dechev, W. L. Cleghorn, J. K. Mills, ”Microassembly of 3D Microstructures Using a Compliant, Passive Microgripper”, Journal of Microelectromechanical Systems, vol. 13, no. 2, April 2004.
[2] H. Asano, et al, ”Path Control Scheme for Vision Guided Micro Manipulation System”, The 11th International Conference on Precision Engineering (ICPE), August 16-18, 2006, Tokyo, Japan
[3] P. Attard, D. Wei, G.N. Patey, and G.M. Torrie, ’The Interaction between Macroparticles in Molecular Fluids.’ J. Chem. Phys. 93, 7360-7368 (1990).
[4] Casal A., Hogg T. and Cavalcanti A., ”Nanorobots as Cellular Assistants in Inflammatory Responses”, in Proc. IEEE BCATS Biomedical Computation at Stanford 2003 Symposium, IEEE Computer Society, Stanford CA, USA, Oct. 2003.
[5] A. Cavalcanti1, et al, Nanorobotics Challenges in Biomedical Applications, Design and Control, IEEE ICECS Int’l Conf. on Electronics, Circuits and Systems, Tel-Aviv, Israel, December 2004
[6] Freitas Jr. R.A., ”Nanomedicine, Vol. I: Basic Capabilities”, Landes Bioscience, 1999.
[7] Y. Sun, B.J. Nelson, ”Microrobotic Cell Injection,” 2001 IEEE Int. Conf. of Robotics and Automation (ICRA2001), Seoul, Korea, May 21-26, 2001, Vol. 1, pp. 620-625.
[8] A. Yurkovic, O. Wang, A. C. Basu, and E. A. Kravitz, Learning and memory associated with aggression in Drosophila melanogaster, PNAS, November 14, 2006; 103(46): 17519 - 17524.
[9] F. L. Lewis, S. Jagannathan and A. Yesildirek, ”Neural Network Control of Robot Manipulators and Nonlinear Systems,” Taylor and Fracis, 1999
[10] M. Shahini, W.W. Melek, J.T.W. Yeow, ”Application of Artificial Neural Network for Friction Compensation during Micro Particle Pushing,” IEEE SMC 2006.
[11] F. L. Lewis, A. Yegildirek, and K. Liu, ”Multilayer neural-net robot controller with guaranteed tracking performance,” IEEE Trans. on Neural Networks, vol. 7, no. 2, pp. 388-399,1996
[12] Y. Tang, F. C. Sun, Z. Sun, and T. Hu, ”Tip Position Control of a Flexible-Link Manipulator with Neural Networks,” International Journal of Control, Automation, and Systems, vol. 4, no. 3, pp. 308-317, June 2006
[13] J. Dayhoff, ”Neural Network Architectures: An Introduction”, Van Nostrand Reinhold, New York, 1990
[14] Y. H. Kim, F. L. Lewis, ”Reinforcement Adaptive Learning Neural-Net-Based Friction Compensation Control for High Speed and Precision,” IEEE Transaction on Control Systems Technology, Vol. 8, No. 1, Jan. 2000, pp. 118-126

[15] D. W. Chung, S. H. Yang, ”Stick-Slip Friction Compensation for Mechatronic Servo Systems,” Proceeding of The 23rd IASTED Conference on Modelling, Identification, and Control, Grindelwald, Switzerland, 23-25th Feb. 2004, pp. 611-615