2002 IEEE International Symposium on Computer
Aided Control System Design Proceedings
September 18-20, 2002. Glasgow, Scotland, U.K.

NEURAL MOTION AND EVOLUTIONARY DECISION IN ROBOTIC COMPETITION APPLIED FOR MOLECULAR MACHINE SYSTEM DESIGN

Adriano Cavalcanti

DARMSTADT UNIVERSITY OF TECHNOLOGY / COMPUTER SCIENCE DEPARTMENT
Fraunhoferstrasse 05, Darmstadt D-64283 Germany / adasilva@igd.fhg.de
Telephone +45+6151+155409 Fax +45+6151+155499

ABSTRACT

The author presents a new approach within advanced graphics simulations for the problem of nano-assembly automation and its application for medicine. The problem under study concentrates its main focus on nanorobot autonomous control for molecular assembly manipulation and the use of evolutionary competitive agents as a suitable way to warranty the robustness of any proposed model. The presented paper summarizes distinct aspects of some techniques required to achieve a successful nano-planning system design with a 3D visualisation in real time.

Keywords: Nanorobotics, Nanosystems, Nanomedicine, Biomolecular Assembly Automation, NanoCAD, Virtual Reality, Physically Based Simulation, Artificial Neural Networks, Genetic Algorithms, Adaptive Control.

1. INTRODUCTION

Nanotechnology is a very new field comprised of an interdisciplinary set of sciences, such as computer science, physics, chemistry and biology. Its impact in our actual society could not be completely and precisely foreseeable, but the only thing that is a common sense is that its possibilities and future results indicate it as an exciting new research branch. Its potential fields of application ranges from the development of new materials in the field of metallurgy to advanced molecular machine systems in the field of medicine. Considering the large variety of aspects that comprises nanotechnology, this paper is presenting the design and simulation of an autonomous mobile robot at atomic scales to perform molecular assembly manipulation for nanomedicine [12]. Robotic manipulation and assembly of objects at the nanoscale is a branch of nanorobotics that has generated considerable interest and promises to produce revolutionary advances in miniaturization. Practical approaches for nano-planning systems has been presented [24] as a first step towards automating assembly tasks in nanorobotics, where was presented a 2D assembly tasks automation. Arguments on the appliance of artificial intelligence as the most appropriate means to enable some aspects of intelligent behaviours for the control of nanorobots, which intends to facilitate a major improvement in the cost-effectiveness of molecular manufacturing finding a suitable assembly sequence for end-specified molecules, has been discussed and accepted in the nano community [09]. In this aspect an acceptable approach is the use of agents as assemblers, where the most suitable model would be projected ideally as close as possible of concepts related to Artificial Life, thus nanoscale assemblers to be useful have to be controlled by robust, scalable, flexible software, which will enable the system to survive in very chaotic environments, and such characteristics could be better satisfied by the use of concepts like self-modifying code, adaptive control systems, ants, and genetic algorithms.

Theoretical work in molecular manufacturing has emphasized the need for very small and very accurate manipulators that simultaneously have a wide range of motion to enable the task of assembly molecular components [11]. New approaches for nanorobotic motion and control design has been proposed, where the models consider thermal noise as a significant source of positional uncertainty, comparing a robotic arm, Stewart platform and a five-strut crank model [26]. A precursor work for nanoassembly automation was presented for a modern molecular library for proteins, DNA, and RNA assembled by highly automated robotic equipment [10] with polymers approaching 10^{14} sequences and libraries with over than 2×10^{5} members [19]. The design of a molecular library by using 160,000 reactions has used a genetic algorithm approach consisted of coding 10 isocyanides, 40 aldehydes, 10 amines, and 40 carboxylic acids in a "bit-string" data structure [35], where was suggested that the method could be fully automated with robotic handling and fluidic transport. More recent work in the possible automation of nanoscale manipulation has produced a fully autonomous motion manipulator system capable of performing 200,000 accurate measurements per second at the atomic scale [25]. In the sense to enable the nano-automation systems prototyping will be addressed in the present work the use of computer graphics for the model design simulation and visualization, thereafter is presented theoretical analysis and mathematical results supported through experimental simulations.

2. NANO ASSEMBLY AUTOMATION

One of the main directions of research in the field of mechatronics is the miniaturization of robots, machines and devices becoming more and more smaller where this new branch is called as micro/nano mechatronics. With

the use of Scanning Probe Microscopes (SPM) as in Atomic Force Microscopy and Scanning Tunneling Microscopy, geometrical and electrical magnetic properties of material can be measured down to atomic scale in 3D with precision at up to 0.01nm resolution. Thus if our main long-range goal is to build an assembler, then we have to do more than speculate on its capabilities but also to describe some of the main aspects needed to make molecular-scale machines. The use of classical rigid-body dynamics and semiclassical mechanics are quite sufficient for studying the rotational dynamics for building molecular components [30], what could be proved through the use of concepts provided by chemistry, protein engineering and scanning probe methods. The molecules will arrange themselves according to their configuration and the temperature of the surrounding medium, therefore we can see that a specific molecular reaction will take a place regarding thermodynamics perspectives and chemical kinetics. The appliance of chemical kinetics for the study of molecular collisions from beam-type experiments can be used to deduce the mechanism of chemical reactions, and from an understanding of the mechanisms we can design specific molecular structures, as we can expect to do for molecular nanotechnology. About the consideration on quantum chemistry, advanced mechanics simplifies the equations of motion by the Langragian and Hamiltonian forms [34] and these are the mathematical methods for quantum mechanics. The major strength of molecular mechanics is that energy minimizations for large systems can be computed in a reasonable amount of time. The computed answers are usually good, although the force constants from the preceding relations were determined by empirical methods. The essential methods of molecular dynamics have been discussed [16], in this manner the kinematics of hard-sphere and soft-sphere collisions could be computed along with the intermolecular potential and time dynamics. The methods usually involve finite-difference computations, which consist of solving partial differential equations. All those concepts and calculations could be found at molecular-design programs like Alchemy III and HyperChem [18]. More practically the quantum mechanical calculations are usually approximated by various methods, including perturbation methods, Huckel methods, and group theory involving symmetry operators, which allow one to achieve reasonable numerical values [30]. Molecules and atoms are generally considered for such mechanics calculations as hard spheres.

Building patterns and manipulating atoms with the use of SPM has been used with satisfactory success [29] as a promising approach for the construction of nanoelectromechanical systems (NEMS), however these manual manipulations require much time and once that is a repetitive task, it tends to be a kind of monotonous and imprecise task when performed manually for a large number of molecules, therefore automation systems in such issue would be greatly improving the productivity and precision in the sense of atoms manipulation. The most important challenge and that has become evident as a vital problem for the nanotechnology fast development with its industrial application is the automation of atoms manipulation. The starting point of nanotechnology to achieve the main goal of build systems in nanoscale is the development of autonomous molecular machine systems, which could enable a manufacturing schedule of nano-device building blocks. The scheduling problem considered here by the author keeps its main focus on nanomedicine, where the biomolecular assembly manipulation is automatically performed by some agent which try to improve the nutritional state of some organism through the injection of appropriate assembled substances into the pre-established delivery points.

3. NANOMEDICINE

The principal focus in medicine is going to shift from medical science to medical engineering, where the design of medically-active microscopic machines will be the consequent result of the techniques provided from human molecular structure knowledge derived during the 20th and the beginning of the 21st century. For the feasibility of such achievements in nanomedicine two primary capabilities are required: fabrication of parts and assembly of parts. Through the use of different approaches such as biotechnology, supramolecular chemistry, and scanning probes, both capabilities had been demonstrated in limited fashion as early as 1998 [12]. Despite quantum effects which impose a relative uncertainty to electron positions, such objections are resolved by recognizing that an atom has a predictably position due its nucleus great mass. The quantum probability function of electrons in atoms tends to drop off exponentially with distance outside the atom, giving atoms a moderately sharp "edge". Even in most liquids at their boiling points, each molecule is free to move only ~0.07 nm from its average position [13]. Recent developments in the field of biomolecular computing [01] have demonstrated positively the feasibility of processing logic tasks by bio- computers [15], which is a promising first step to enable future nanoprocessors with increasing complexity, power of information storage and data processing capacity, which could be considered as an indispensable devise for a real autonomous nanosystem. Other developments in the sense of building biosensors [33] and nano-kinetic devices [32] have advanced recently too, and could be considered for many researchers as a prerequisite for making nano-automation feasible enabling nanorobotics operation and locomotion. Many classical objections related to the feasibility of nanotechnology, such as quantum mechanics, thermal motions and friction, have already been considered and resolved [30]. Natural molecular machine systems could be found operating in living things: in the human body the most similar with a

molecular assembler is a ribosome. A ribosome acts as a general purpose factory building diverse varieties of proteins by bonding amino acids together in precise sequences under instructions encoded in the DNA. Similarly a pre-established set of assembly tasks will be performed by the presented nanorobot with such molecular manipulation performed by a *telescoping manipulator* device [11].

4. PROPOSED DESIGN

Molecular machine systems could be described as a system capable to perform molecular manufacturing at atomic scale. Actually the three main design approaches in nano manipulation for the liquid and air environment are: robotic arm, Stewart platform and a five-strut crank model; for our experiments we chose a nano-manipulation in a liquid environment, which is most relevant within the presented application in nanomedicine. It was demonstrated that computation is relatively cheap for macroscale robotic actuators while arm motion is relatively cheap for nanoscale robotic actuators. Thus the moment-by-moment computer control of arm trajectories is the appropriate paradigm for macroscale robots, but not for nanoscale robots [12]. For nanoscale robots, the appropriate manipulator control is often trajectory trial and error, also known as sensor based motion control [20]. The model described was developed using OpenGL [37] and C++ [21]; for the input and user interface was adopted the mouse and keyboard, and the camera view can also change its position in the y-axis related to the user's view height (figure 1).

4.1. Virtual Environment

There is a general agreement about the importance and necessity of the use of advanced graphical simulation that can accurately reflect the results of experiments in NanoCAD and automated planning to the judgments about manufacturing feasibility assisting chemical and biological assembly analyses in nanotechnology. Nanoscale object manipulation systems have been successfully applied with the use of computer graphics for the teleoperation, where the requirements for such systems have been clearly established [31]. Virtual Reality was used for our nanorobot design where the use of macro and microrobotic concepts is considered as a practical approach once the theoretical and practical assumptions here have focused on its domain of appliance. The design should be robust enough to operate in an environment with movements in six-degrees-of-freedom.

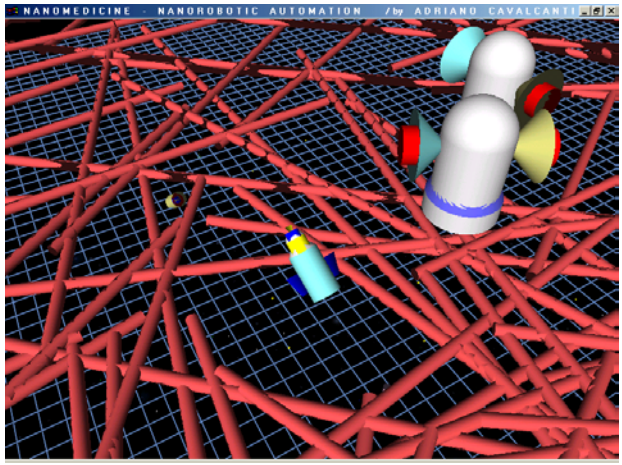
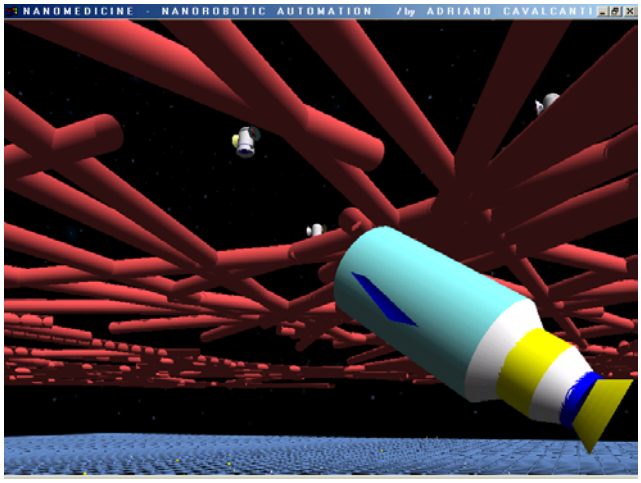


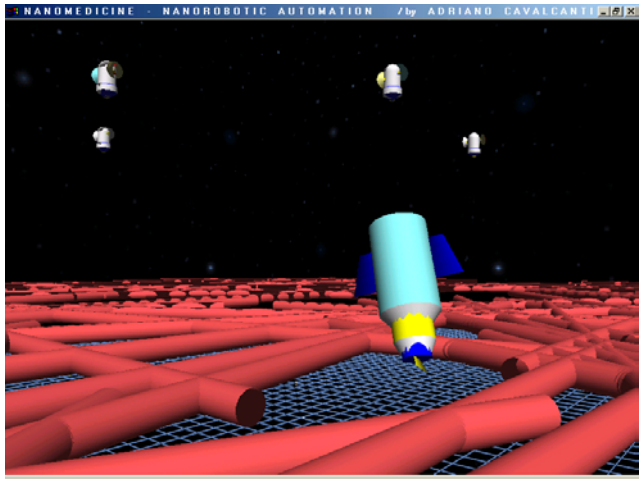
Figure 1. Top camera view in the virtual environment.

Therefore a suitable starting point for our hypotheses formulations and autonomous assembly system experiments was to consider the nanorobot design derived from biological models and comprised of some basic nanoscale components such as *molecular sorting rotors* and a *telescoping manipulator* [11], meanwhile for the nanorobot design was assumed concepts provided from underwater robotics [36]; for the kinetics assumptions the nanorobot lives in a world of viscosity, where friction, adhesion, and viscous forces are paramount and gravitational forces are of little or no importance [12]. The main argument to use concepts based on underwater robotics as a good starting point for design is the liquid environment where the agents will be under operation performing the biomolecular assembly tasks. The *telescoping manipulator* will be carried internally inside the nanorobot, and the nanorobot will push the assembled molecule to the delivery point.

The obstacles will be located in unknown probabilistic positions (figure 2). The delivery positions that represent organ inlets requiring proteins to be injected are located in a well-known position for the nanorobot. The trajectories and position of each molecule were generated randomly and each one will have also a probabilistic motion acceleration, thus a new trajectory is generated dynamically by using some collision each time that a molecule has received an impulse or collided with an obstacle. The nanorobot navigation uses plane surfaces (three fins total) and bi-directional propellers, which is comprised of two simultaneously counter-rotating screw drives for the propulsion [12]; the nanorobot has sensors which will inform if a collision happens and either if it is an obstacle to generate a new trajectory planning, or if it could be a molecule which has to be captured and assembled.



(2.a) Avoiding obstacles



(2.b) Finding path

Figure 2. Robot collision detection.

In the simulation while some molecules have been captured (figure 3) the robot arm in the proposed nanorobot will manipulate other molecules internally, afterwards they will be delivered in the desired positions.

The nanorobot will live in a world dominated by viscosity as well as bacteria do. In this world a very low Reynolds number (Re) is assumed for the kinetic calculations [28], where the fluid mechanics in small structures can usually be described by the classical continuum equations [11]. The ratio of inertial to viscous forces is determined by Re which could be expressed in equation 01.

$$\mathrm{Re} = r\rho v / \eta \qquad (01)$$

where η is the viscosity of the fluid, v is the velocity, ρ is the fluid density, and r is a characteristic dimension or fluid density. Re indicates whether the flow will be laminar or turbulent [08] around an object of a given shape. For nanoscale dimensions in fluids of ordinary viscosities and velocities, Re is low and the flow laminar [12]. The inertial force on the object is of order $F_{inertial} \cong rv^2L^2$ and the viscous drag force is of order $F_{viscous} \cong hvL$. Thus in order to keep moving forward a nanorobot of size L $\cong 1$ micron and velocity $v \cong 10$ micron/sec must apply $F_{inertial} \cong 10^4\, fN$ (femtonewtons, $1fN$ = 10^{15} N) and a much larger $F_{viscous} \cong 10\, fN$ of motive force. For instance, if motive power to a swimming nanorobot with radius $\mathrm{Re}_{nano} = 1$ micron, and the velocity $V_{nano} = 1$ cm/sec, is suddenly stopped, then the nanorobot will "coast" to a halt in a time $t_{coast} = r\,\mathrm{Re}_{nano}$ and by equation 2:

$$t_{coast} = \frac{r\,\mathrm{Re}_{nano}^2}{15\eta} = 0.1 \qquad (02)$$

where 0.1 is expressed in microsecond, and in distance $x_{coast} \cong v_{nano}t_{coast} = 1\,\mathrm{nm}$ [05]. Thus with n as the rotational frequency, if the nanorobot is rotating at a frequency $n_{nano} = 100$ Hz when its rotational power source is suddenly turned off, n_{nano} decays exponentially to zero in a time $t_{coast} \cong 0.1$ microsecond and stops after turning, as expressed by equation 03:

$$q_{coast} = \frac{2pn_{nano}r\,\mathrm{Re}_{nano}^2}{15\eta} \cong 40 \qquad (03)$$

where q is the rotational motion, p is the pressure and 40 is expressed in microradians.

4.2. Physically Based Simulation

The study of non-penetrating rigid bodies in virtual reality for dynamic constrained simulation is a field of research in computer graphics that has an enormous impact for physically based simulation and a large range of works in this field have produced good achievements. Particularly in calculating motions of many objects that move under changing constraints and frequently make collisions, one of the key issues of dynamic simulation methods is calculation of collision impulse between rigid bodies. The correlation between contact force and relative normal acceleration could be expressed as a linear programming problem [03], what permits to calculate the collision impulse that works between rigid

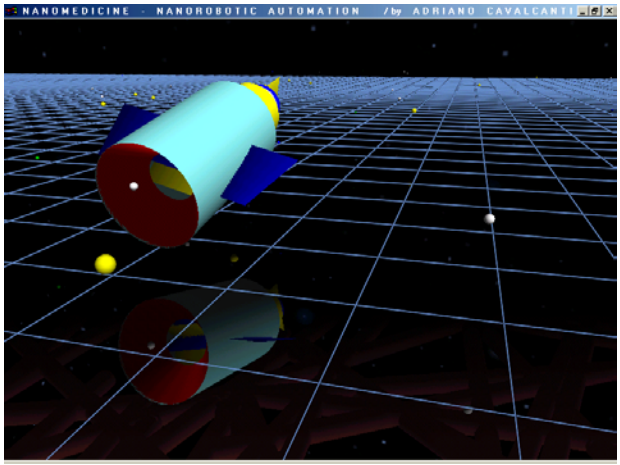


Figure 3. Molecular identification by collisions contact.

bodies colliding at multiple points. Furthermore the relation between collision impulse and relative normal velocity could be also expressed as a linear complementary problem. A simple and fast algorithm for calculating contact force with friction by formulating the relation between force and relative acceleration as a linear complementary problem was equally demonstrated [04], and this model was based on Dantzig's algorithm or solving linear complementary problem, which is extended for systems with friction. Baraff's algorithm has achieved great performance for real-time and interactive simulation of two-dimensional mechanisms with contact force, friction force and collision impulse, although friction impulse at collision was not completely covered in such model. Therefore was established a complementary algorithm covering as well the "impulse-based" aspects, which can trace in detail the change of friction force at a single colliding point by numerical integration of both contact force and friction force [27]. In the physical world, there are no perfectly planar faces or perfectly straight edges, and specifically at a nanoscopic level all contacts can be modelled as a composition of point contacts. Dynamic collision detection and non-penetrating constraint assumptions was indispensable for the model development once that we are going to consider kinetics and frictional aspects required specially for molecular assembly manipulation and rigid body motion with hydrodynamics at low Re [12].

Basically the problem of collision detection corresponds to determining whether there is any contact between two objects. We can express the exact conditions for dynamically contact forces as a vector C of contact force magnitude, which is correct if it satisfies some of the basic conditions discussed next. There is no object interpenetration through contact forces for rigid body, and any contact force can only push any related object. The contact force could not be used to pull any 3D object, it affects just the contact points and anything else otherwise. For dynamic collision detection the contact force express a continuous behaviour related to the function of time. Such assumptions are necessary for any correct contact force function that intends to produce a dynamically correct motion. It is possible to happen multiple correct contact force and when some similar circumstances arises the right solution is given using an equation of compatibility, what is precluded by the rigid body assumption, nevertheless any correct result provided by the contact force C result in the same correct motion [03]. The motion of a rigid body subject to external forces is described by the Newton-Euler motion equations as follows:

$$\dot{v} = \frac{1}{m}\sum_{i=1}^{k} f(C)_i \tag{04}$$

$$\dot{w} = I^{-1}\left(\sum_{i=1}^{k} C_i \times f(C)_i - w \times Iw\right) \tag{05}$$

where $\dot{v}$ is the dotted velocity vector, $\dot{w}$ is the dotted normal contact distance vector, $f(C)_i$ are the external forces (including contact forces), C_i are the vectors which point from the center of mass to the points where the force apply, **I** denotes the inertia tensor, and m the object mass. We are interested to verify when the objects begin to their motion if there is any contact between the objects. For rigid body simulation there are two types of contacts [23] that we could identify as tangential collision and boundary collision.

Tangential Collisions: this corresponds to a tangential intersection between two surfaces at a geometric contact point. The contact point lies in the interior of each surface and the normal vectors at that point are collinear. Equation 06 expresses a tangential intersection.

$$E(s,t) = P(u,v) \tag{06}$$

$$(E_s(s,t) \times E_t(s,t)) \bullet P_u(u,v) = 0 \tag{07}$$

$$(E_s(s,t) \times E_t(s,t)) \bullet P_v(u,v) = 0 \tag{08}$$

with E(s,t) and P(u,v) representing two parametric surfaces, we assume that the Bézier surface has an algebraic formulation in homogeneous coordinates as:

$$E(s,t) = (X(s,t), Y(s,t), Z(s,t), W(s,t)) \tag{09}$$

$$P(u,v) = (\overline{X}(u,v), \overline{Y}(u,v), \overline{Z}(u,v), \overline{W}(u,v)) \tag{10}$$

where E_s, E_t, P_u, P_v correspond to the partial derivatives and $\bullet$ corresponds to the dot product. Equation 06 corresponds to a contact between the two surfaces; equation 07 and 08 represent the fact that their normals are collinear. They are expressed as scalar triple product of the vector. This is an over constrained system and has a solution only when the two surfaces are touching each other tangentially.

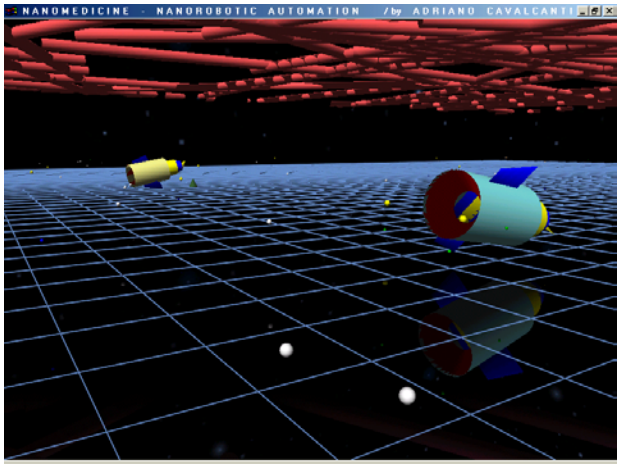


Figure 4. Competitive agent and reagent in action.

For such equations, after cross multiplication we get 3 polynomial equations of degree 2n each. The dot product results in the addition of degrees of the numerator polynomials. Similarly for two algebraic surfaces, the problem of tangential intersection can be formulated as:

$$e(x,y,z) = p(x,y,z) = 0 \qquad (11)$$

with

$$\begin{pmatrix} e_x(x,y,z) \\ e_y(x,y,z) \\ e_z(x,y,z) \end{pmatrix} = \theta \begin{pmatrix} p_x(x,y,z) \\ p_y(x,y,z) \\ p_z(x,y,z) \end{pmatrix} \qquad (12)$$

Equation 11 and 12 correspond equally to an over constrained system.

Boundary Collisions: this intersection lies on the boundary curve of one of two surfaces. Thus given a Bézier surface, defined over the domain, $(s,t) \in [0,1] \times [0,1]$, we obtain the boundary curves by substituting s or t to be 0 or 1. The resulting problem reduces to solving the equation:

$$E(s,1) = P(u,v). \qquad (13)$$

Two objects collide if equations 06 or 13 for parametric surfaces and the 06 for algebraic surfaces have a common solution in their domain.

4.3. Competitive Evolutionary Decision

We intend to construct and validate a nano-planning system, where through the use of competitive evolutionary agents shall enable a better tuned validation of our autonomous nanorobot system under study, thus they compete against each other (figure 4) in the sense that meanwhile one agent try to improve the nutritional

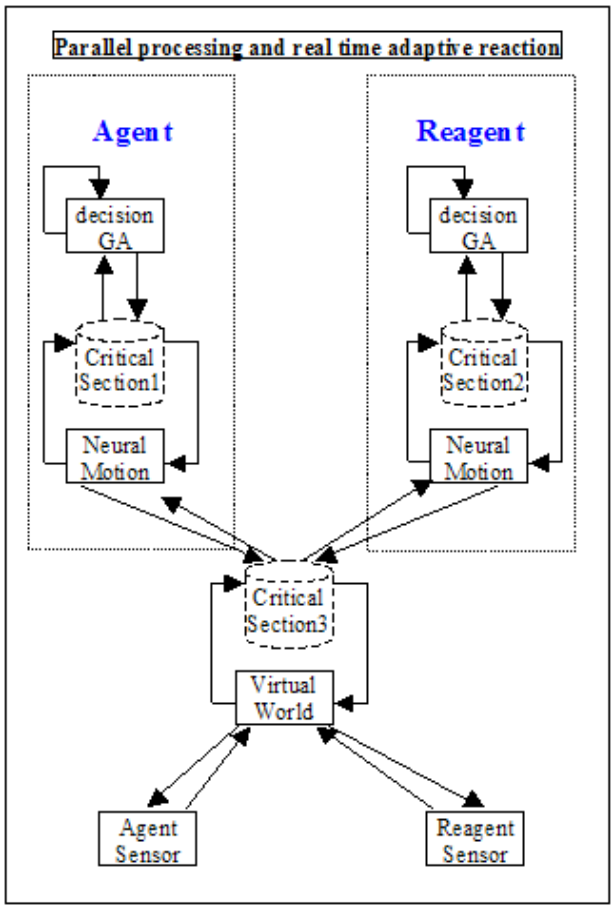


Figure 5. System architecture.

state of a set of organ inlets in the represented living three-dimensional environment, the reagent try to debilitate it through the injection of inappropriate assembled substances into the same organ inlets. Autonomous behaviour in complex real-world systems requires accurate and timely reactions to environmental events, thereby advances in artificial intelligence and real time systems have become important and successfully tools leading with such problems, and the use of concepts derived from Evolutionary Techniques, Artificial Life and Ants has received a special attention in the research community. The evolutionary model used for the nanorobot autonomous decision is cited in the literature as Genetic Algorithms (GA). GA relies on concepts derived from evolution and genetics [06], such as mutation, crossover, chromosome selection, adaptability, thus providing a behavioural learning with events and actions through time. In a GA every solution is seen as an individual with its own genetic characteristics and belonging to a certain population. Given a set of tentative solutions, only the better-adapted ones will have a proportionally greater probability to be selected to reproduce via crossover and to generate new solutions, thus perpetuating their genetic information. These new individuals tend to be better fitted than their parents, which lead to the conclusion that after several

generations the population will be composed of highly adapted individuals since the worse solutions were replaced during the evolutionary process. Mutation acts as a diversity increment tool because after some generations, the population tends to lose diversity. In the implemented architecture (figure 5) we used real time and parallel processing techniques [38], which was intended to provide a simulation scenery as close as possible of a real situation, where both agents react adaptively to any action performed by its adversary decision with the model visualization in real time [07].

Each solution in the GA model is described as a chromosome describing the agent decision on how, when and what organ inlets to be attended, what is detailed next.

$$Max \quad f(r_\Omega) = \sum_{t=1}^{n}\sum_{i=1}^{m} \psi w_i^t - \left| y^t \right| \tag{14}$$

s.t.
$$y^t = Q^t - d \tag{15}$$
$$Q^t = \sum x_i^t \le L \tag{16}$$
$$x_i^t = \mu_i^t \, x_i^{\max} \tag{17}$$
$$\mu_i^t \le \Delta_i^{\max} \tag{18}$$
$$w_i^{t+1} = w_i^t - \gamma\psi z_i^t + \psi x_i^t \tag{19}$$
$$w_i^{\min} \le w_i^t \le w_i^{\max} \tag{20}$$
$$\mu_i^t \in \{\{0,100\} \vee \{0,1\}\} \tag{21}$$
$$\Omega \in \{A, B\} \tag{22}$$
$$\psi \quad \begin{cases} \Omega = A \Rightarrow \psi = 1; \\ \Omega = B \Rightarrow \psi = -1; \end{cases} \tag{23}$$

where

r, t, i: subscript denoting: robot, time, organ inlet.
max, min: upper and lower bound parameter.
A, B: agent and reagent respectively.
n: size of time in the simulated scenery.
m: total of organ inlets to be fed.
L: robot load capacity.
y^t: surplus/deficit to the desired assembled mean.
x_i^t: substance amount injected in the organ inlet *i*.
Q^t: total assembled molecule by *r* in *t*.
w_i^t: chemical state of the organ inlet *i* at time *t*.
z_i^t: adversary substance injected to organ inlet *i*.
d: desired assembled substances rate.
γ: parameter to look ahead nutritional levels.
μ_i^t: boolean variable.
ψ: determines the kind of behaviour for *r*.
Ω: determines if *r* is agent or reagent.
Δ: maximum to be injected at organ *i* in *t*.

Equation 14 represents our fitness function, where the agent maximize the protein levels for the selected organ inlets, meanwhile the reagent minimizes the same parameter, and the variable *y* induces the nanorobot to catch a number of molecules as closely as possible to the desired delivery mean. Equation 15 sets up the specified amount to be transported and assembled at time *t* for the nanorobot. Equation 16 is the total sum of captured molecules that will be assembled attending the nanorobot load capacity. Equation 17 is the amount specified for each organ inlet *i* with injection at time *t*. Equation 18 expresses the maximum that could be injected in the organ inlet *i* at time *t*. Equation 19 is the nutritional state for the organ inlet *i* due to the action performed by *r* and its adversary. Equation 20 sets the minimum and maximum nutritional levels desired for the organ inlets. Equation 21 is the genetic random operating values. Equation 22 defines if *r* is agent or reagent. Equation 23 determines specific model performance for *r*. The competitive nanorobot interactive rule is described in the table 1, where e, g and h represent the kind of molecule to be assembled by *r*, therefore*:*

$$\beta \quad \begin{cases} r_\Omega = A \Rightarrow \beta = e, \\ r_\Omega = B \Rightarrow \beta = h, \end{cases} \tag{24}$$
$$\delta = g. \tag{25}$$

The *min* denotes the minimum defined to be captured by each nanorobot at time step t. As we are going to see, the action based on sensor local perception has generated a competitive nanorobot coherent behaviours, which was observed by the proposed model simulation. The study of autonomous multi-robot behaviour in a single global environment is a field of research relatively new [22], which has advanced the most of concepts related to the use of local perception for reactive agents.

Step 1: r_Ω walk randomly to capture β and δ;
Step 2: if ∑ β = ∑ δ → assemble f(r_Ω)= β + δ;
Step 3: if ∑ f(r_Ω) < min repeat step 1;
Step 4: r_Ω achieve next delivery goal regarding the delivery queue;
Step 5: if $\Omega = B$ go to step 7, otherwise next step;
Step 6: if delivery_NOT_overdose = true → next step; otherwise go to step 8;
Step 7: delivery: f(r_Ω) = f(r_Ω) -1;
Step 8: if f(r_Ω)>0 repeat step 4;
Step 9: repeat step 1;

Table 1. Competitive nanorobot interaction rule.

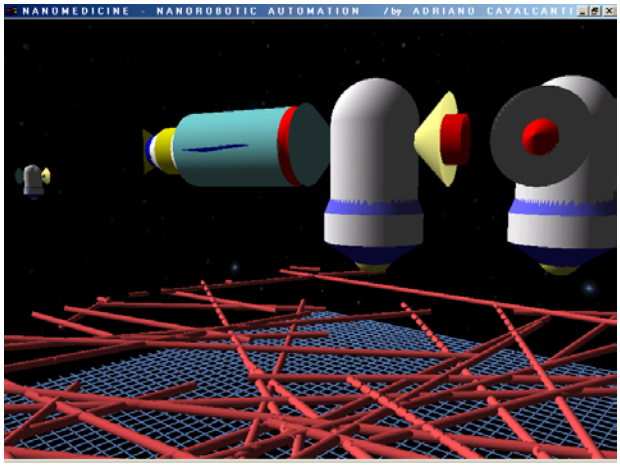


Figure 6. Nanorobot molecule delivery to the organ inlet (represented by the white cylinder).

The proposed nanorobot model is not leading with any kind of nanorobot self-replicating behaviour, instead of it the model uses an evolutionary approach strictly for the combinatorial analyses, thus the nanorobots react adaptively in an uncertain environment with a well defined pre-programmed set of actions for the biomolecular assembly task.

4.4. Neural Motion Control

A connectionist model using artificial neural network (ANN) was chosen for the solution of motion control and shortest-path problem, where we are going to lead with a dynamic combinatorial problem for each time-step simulation. The classical problem of finding an optimal three-dimensional shortest path avoiding polygonal obstacles was demonstrated as typical NP-hard [02]. The use of a non-deterministic approach to solve the motion control seems to be the appropriate technique in such cases, in the sense that among other heuristic methods the use of ANN was successfully used for motion and animation of physically-based models in virtual environments [14]. In our case we have implemented a feedforward or acyclic network due to its suitability for probabilistic calculations. The model particularly implemented here is known as a Neural Sigmoid Belief Network (NSBN) [17], which requires a lower computational effort in comparison with a backpropagation approach. The properties of a NSBN could be described by equation 26.

$$pa(X_j) \subseteq \{X_1, X_2, ..., X_{j-1}\} \qquad (26)$$

where X represents a vector, consisting of the two-valued random variables *X1, X2,..., Xn*, defining a topology composed of *N* stochastic neurons.

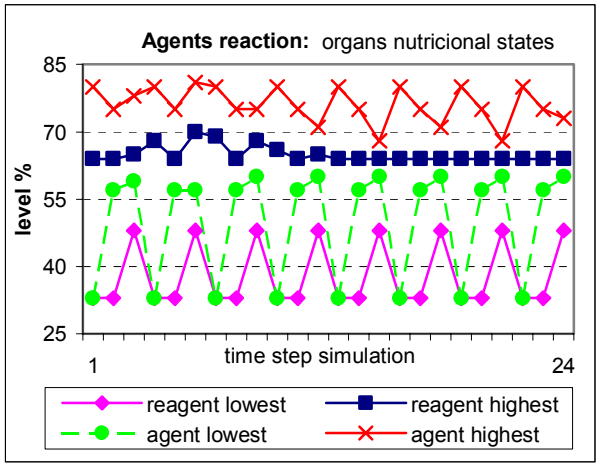


Figure 7. Highest/lowest organ inlets nutritional levels.

With *n* representing the range of hidden layer, which leads the network to be optimized at the time-step *t*, it related to each destiny to be achieved for each agent through the simulation. The units in the network are organized into a two-dimensional *n* rows by *m* columns matrix A_{mn}, where *n* and *m* is the cost matrix of destinations to be performed by each evolutionary nanorobot, which tries to complete its set of tasks successfully as fast as possible. Let the output of the unit in row *i* and column *j* be $v_{ij} = 1$, where $i \neq j$. This means that the referred destiny is visited at the i^{th} stop, with $v_{ij} = 0$ otherwise. Therefore, a solution cost for each agent routing could be expressed by equation 27.

$$\min R^t = \sum_i \sum_j v_i w_{ij} \qquad (27)$$

The best solution was given by running our simulation based on the distance from each intended goal in the virtual environment configuration (figure 6). A NSBN pseudo code is described at table 2.

5. SIMULATION AND CONCLUSIONS

The present work has intended either to elaborate an advanced three-dimensional graphic environment using neural motion and physically based simulation applied to nanorobotics automation and nanosystems design for nanomedicine, as well as to postulate the use of competitive agents as a systematic way to verify the model robustness within real-time control constraints under a large range of uncertainty.

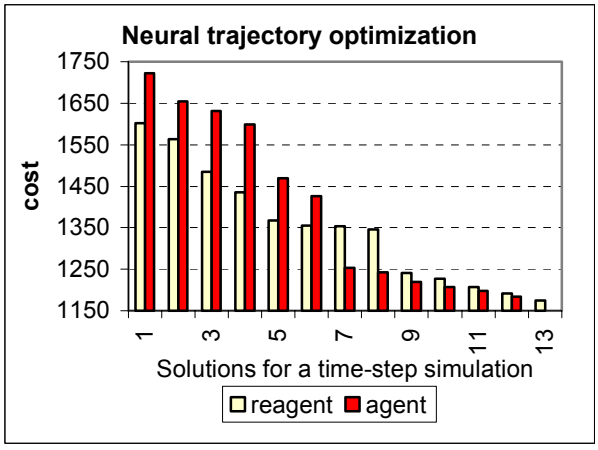


Figure 8. Number of steps for each nanorobot r find the better solution at time i.

A coherent competitive behaviour with a fast adaptive reaction was suitably achieved as it could be demonstrated. The parameter organs' nutritional level in the simulation was initialized at 65% of each relative organ capacity. An ideal performance could be considered as a situation where all nutritional levels have ranged between 30% and 70%. We have established as a critical nutritional situation for instances with values lower than 10% or higher than 90%, which would be represented as a deficiency or an overdose case respectively. The implemented model has generated satisfactory performances with most of organs' nutritional levels floating around 55% of their capacity (figure 7), where just a few levels were a bit higher or lower but it never surpassing 81% or falling under 33%, which indicates no overdoses or deficiencies of the organs' nutritional levels.

Table 2. NSBN pseudo code.

```
timeSeconds=Φ;
time_begin = time(NULL);
do{//Generate NSBN Solutions
j=0;
for(move=0;move<nDestiny;move++)
{  neuronActiv=randomLayer(nDestiny-move);
   // Take the activated neurons.
search.sequence[j]=neuronSelect[neuronActiv];
  for(i=neuronActive;i<(nDestiny-move)-1;i++)
  {  neuronSelect[i]=neuronSelect[i+1];
  }
   j++;
}
// Compare the actual cost and take this
// solution if it has the best cost.
reckonNeuralCost();
time_end = time(NULL);
}while(time_end - time_begin < timeSconds);
```

The nanorobot has required a motion control model based on either of main aspects: optimization of the trajectory distance, and real time analyses for a required trajectory which enable the delivery of assembled biomolecules with avoidance of obstacles. The use of Artificial Neural Networks has demonstrated to be a suitable approach for the nanorobot motion also in a virtual environment with 6-degrees-of-freedom. Thus the neural motion control has achieved suitable results (figure 8) with a low processing requirement and providing shortest-path values till 31,24% better than a greedy solution for the route distance minimization.

As it has been demonstrated the main proposed aspects in the presented work was successfully fulfilled, which indicates that the discussed approach could be a promising system design for a fast automation and prototyping in the nanotechnology development.

6. ACKNOWLEGDEMENTS

The author thanks Robert A. Freitas Jr. for helpful comments on an earlier version of this paper.

7. REFERENCES

[01] L. M. Adleman, "On Constructing a Molecular Computer", DNA Based Computers, 1995, http://olymp.wu-wien.ac.at/usr/ai/frisch/local.html .

[02] T. Asano, D. Kirkpatrick, C. K. Yap, "D1-optimal motion for a rode", Proc. of the 12th Annual Symposium on Computational Geometry, New York, ACM Press, 1996, pp. 252-263.

[03] D. Baraff, "Analytical methods for dynamic simulation of non-penetrating rigid bodies", in Computer Graphics Proceedings, ACM SIGGRAPH, 1989, vol. 23, pp. 223-232.

[04] D. Baraff, "Fast contact force computation for nonpenetrating rigid bodies", in Computer Graphics Proceedings, Annual Conf. Series. ACM SIGGRAPH, 1994, pp. 23-34.

[05] H. C. Berg, "Dynamic properties of bacterial flagellar motors", Nature 249, may 1974.

[06] A. Cavalcanti, "A Virtual Environment for Evolutionary Autonomous Optimization of Real Time Stochastic Control Design", Proc. of IEEE Int'l Conf. on Information, Decision and Control, Adelaide, Australia, 2002, pp. 83-88.

[07] A. Cavalcanti, "Parallel Processing Applied for Scientific Visualization of Industrial Adaptive Control Simulation", Proc. of Int'l Conf. on Computer Graphics and Imaging, IASTED, Honolulu, USA, 2001, pp. 258-259.

[08] A. Cavalcanti, T. Ohishi, A. S. Mendes, F. A. Franca, E. A. R. Delgado, "Using Genetic Algorithm and Simplex Method to Stabilize an Oil Treatment Plant Inlet Flow", Proc. of Int'l Pipeline Conf. 2000, ASME, Calgary, Canada, 2000, pp. 1459-1466.

[09] A. Czarn, C. MacNish, "From Nanotechnology to Nano-Planning", The 9th University of Western Australia Computer Science Research Conf., Nedlands, Western

Australia, Department of Computer Science, The University of Western Australia, 1998, 1:pp 73-85.

[10] S. H. Dewitt, A. W. Czarnik, "Combinatorial Organic Synthesis Using Parke-Davis's DIVERSOMER Method", Accounts Chem. Res., 29, 1996, pp. 114-122.

[11] K. E. Drexler, "Nanosystems: molecular machinery, manufacturing, and computation", Wiley & Sons, 1992.

[12] R. A. Freitas Jr., "Nanomedicine", Volume 1: Basic Capabilities, Landes Bioscience, 1999, http://www.landesbioscience.com/nanomedicine .

[13] A. Goldbeter, "Biochemical Oscillations and Cellular Rhythms: The Molecular Bases of Periodic and Chaotic Behavior", Cambridge University Press, N.Y., 1996.

[14] R. Grzeszczuk, D. Terzopoulos, G. Hinton, "NeuroAnimator: Fast neural network emulation and control of physics-based models". In M. Cohen, ed., Proc. of ACM SIGGRAPH 98 Conf., 1998, pp. 142-148.

[15] M. Hagiya, "From Molecular Computing to Molecular Programming", Proc. 6th DIMACS Workshop on DNA Based Computers, held at the University of Leiden, Leiden, The Netherlands, 2000, pp. 198-204.

[16] J. M. Haile, "Molecular Dynamics Simulations, Elementry Methods", John Wiley and Sons, New York, 1992.

[17] S. Haykin, "Neural Networks a Comprehensive Foundation", 2nd edition, Prentice Hall, New Jersey, USA, 1999.

[18] HyperChem, "Computational Chemistry", Publication #HC40-00-03-00, Hypercube, Inc. Waterloo, Ontario, Canada, 1994.

[19] S. Kauffman, "Random Chemistry", Perspect. Drug Discovery Design, 2, 1994, pp. 319-326.

[20] M. Khatib, B. Bouilly, T. Simeon, R. Chatila, "Indoor Navigation with Uncertainty using Sensor Based Motions", Proc. 1997 IEEE Int'l Conf. on Robotics and Automation, 1997, pp. 3379-3384.

[21] D. Kruglinski, G. Sheperd, S. Wingo, "Programming Microsoft Visual C++", 5th Edition, Microsoft Press, Washington, USA, 1998.

[22] C. R. Kube, "Collective Robotics: from Local Perception to Global Action", PhD thesis, Dept. of Computer Science, University of Alberta, Edmonton, 1997.

[23] M. C. Lin, "Efficient Collision Detection for Animiation and Robotics", PhD thesis, Dept. of Electrical Engineering and Computer Science, University of California, Berkeley, USA, 1993.

[24] J. H. Makaliwe, A. A. G. Requicha, "Automatic planning of nanoparticle assembly tasks", Proc. IEEE Int'l Symp. on Assembly and Task Planning, Fukuoka, Japan, 2001, pp. 288-293.

[25] S. Martel, P. Madden, L. Sosnowski, I. Hunter, S. Lafontaine, "NanoWalker: a fully autonomous highly integrated miniature robot for nano-scale measurements", Proc. of the European Optical Society and SPIE Int'l Symposium on Envirosense, Microsystems Metrology and Inspection, Munich, Germany, Vol. 3825, 1999, pp. 64-76.

[26] R. C. Merkle, "A New Family of Six Degree of Freedom Positional Devices", Nanotechnology, vol. 8, No. 2, 1997, pp. 47-52.

[27] B. Mirtich, J. Canny, "Impulse-based simulation of rigid bodies", in Proc. of Symposium on Interactive 3D Graphics, 1995, pp. 392-398.

[28] M. Ramia, D. L. Tullock, N. P. Thien, "The Role of Hydrodynamic interaction in the locomotion of microorganisms", Biophys. J. 65 (1993):755778.

[29] A. A. G. Requicha, R. Resch, N. Montoya, B. E. Koel, A. Madhukar, P. Will, "Towards hierarchical nanoassembly", IEEE/RSJ Int'l Conf. on Intelligent Robots & Systems, Kyongju, Korea, 1999, pp. 34-39.

[30] E. A. Rietman, "Molecular Engineering of Nanosystems", Biological Physics Series, Springer-Verlag, New York, USA, 2001.

[31] M. Sitti, K. Hashimoto, "Teleoperated Nano Scale Object Manipulation", in Recent Advances on Mechatronics, Springer Verlag Pub., Ed. By O. Kaynak, 1999, pp. 172-178.

[32] R. Stracke, K. J. Böhm, J. Burgold, H. Schacht, E. Unger, "Physical and Technical parameters determining the functioning of a knesin-based cell-free motor system", Nanotechnology 11, UK, 2000, pp. 52-56.

[33] J. Sun, M. Gao, J. Feldmann, "Electric Field Directed Layer-by-Layer Assembly of Highly Fluorescent CdTe Nanoparticles", Journal of Nanoscience and Nanotechnology, Vol.1, No.2, American Scientific Publishers, 2001, pp. 21-27.

[34] K. R. Symon, "Mechanics", 3rd Edition, Addison-Wesley, Reading, MA, 1971.

[35] L. Weber, S. Wallbaum, C. Broger, K. Gubernator, "Optimization of the Biological Activity of Combinatorial Compound Libraries by a Genetic Algorithm", Angew. Chem. Int. Ed. Engl., 34(20), 1995.

[36] L. L. Whitcomb, "Underwater Robotics: out of the research laboratory and into the Field", Int'l Conf. on Robotics and Automation, IEEE, 2000, pp. 85-90.

[37] M. Woo, J. Neider, T. Davis, D. Shreiner, "OpenGL Programming Guide", 3rd Edition, Addison-Wesley, Boston, USA, 1999.

[38] C. Wurll, D. Henrich, H. Wörn, "Parallel on-line Motion Planning for Industrial Robots", 3rd ASCE Specialty Conf. on Robotics for Challenging Environments, Robotics 98, New Mexico, USA, 1998, pp. 308-314.

Adriano Cavalcanti is a member of the technical staff at Fraunhofer Institute for Computer Graphics, where he has been developing works on several research projects related to virtual reality, autonomous systems, parallel processing, and 3D physically based simulation. He holds a B.S. in C.S. from Sao Paulo State University, and a M.S. in E.E. focusing on control automation from Campinas State University, Brazil. He is currently a Ph.D. student in the Computer Science Department at Darmstadt University of Technology, Germany.