Robotic Assembly Replanning Agent Based on Neural Network Adjusted Vibration Parameters

Lejla Banjanovic-Mehmedovic and Senad Karic
Faculty of Electrical Engineering University of Tuzla, H&HInc.
Bosnia and Herzegowina

1. Introduction

The applications of robot are very extended and have already become classic in different branches of mass industrial production such as welding, painting by spraying, antirust protection, etc. Though the operations performed by robots in these fields are very complex, the operations of assembly are even more complex. In fact, robot assembly operations involve the process of direct solving the conflicting situations being not within the classic repetitive work.

Investigations treating typical assembly duties started forty years ago (Bohman, 1994). In the meantime, it was offered a series of control mechanism of mating date. Performing assemblies depends on sensation of and appropriate reaction to the forces of contact between mating components date (Wei, 2001).

It is shown that with the intelligent techniques, example components can be assembled faster, gentle and more reliably. In order to create robot behaviours that are similarly intelligent, we seek inspiration from human strategies date (Chan, 1995). The working theory is that the human accomplishes an assembly in phases, with a defined behaviour and a subgoal in each phase. The human changes behaviours according to events that occur during the assembly and the behaviour is consistent between the events. The human's strategy is similar to a discrete event system in that the human progresses through a series of behavioural states separated by recognizable physical events.

In achieving acceptably fast robot behavior with assuring contact stability, many promising intelligent-control methods have been investigated in order to learn unstructured uncertainties in robot manipulators date (Chan, 1995), (Miyazaki et al., 1993), (Brignone et al., 2001). For example, (Newman et al., 2001) work describes intelligent mechanical assembly system. First phase for assembly is blind search. In this phase multiple parameters are assigned to rotational search attractor. If sensors register force values higher then thresholds, new parameters are assigned. Intelligent layer is represented on 22-dimensional space of trajectories, and based on blind search parameters (correct and incorrect) neural network is made. Correct assembly path is chosen by using form of Genetic algorithm search, so the new vectors are evolved from most successful "parents". Using this process, the robot was allowed to generate and test its own program modifications.

The primary source of difficulty in automated assembly is the uncertainty in the relative position of the parts being assembled (Vaaler, 1991). The crucial thing in robot assembly is how to enable a robot to accomplish a task successfully in spite of the inevitable uncertainties

(Xiao & Zhang, 1995). Often a robot motion may fail and result in some unintended contact between the part held by the robot and the environment. There are generally three types of approaches to tackle this problem. One is to model the effect of uncertainties in the off-line planning process, but computability is the crucial issue. A different approach is to rely on–line sensing to identify errors caused by uncertainties in a motion process and to replann the motion in real-time based on sensed information The third approach is to use task-dependent knowledge to obtain efficient strategies for specific tasks rader than focusing on generic strategies independent of tasks.
(Xiao & Zhang, 1995) introduced a systematic replanning approach which consisted of patch-planning based on contact analyses and motion strategy planning based on constraints on nominal and uncertainty parameters of sensing and motion. In order to test the effectiveness of the replanning approach, they have developed a general geometric simulator SimRep on a SUN SPAR@ Station which implements the replanning algorithms, allows flexible design of task environments and modeling of nominal and uncertainty parameters to run the algorithms and simulates the kinematics' robot motions guided by the replanning algorithms in the presence of uncertainties.
In our paper, we present the complex robot assembly of miniature parts in the example of mating the gears of one multistage planetary speed reducer. Assembly of tube over the planetary gears was noticed as the most difficult problem of overall assembly and favourable influence of vibration and rotation movement on compensation of tolerance was also observed. There were extensive experimental complex investigations made for the purpose of finding the optimum solution, because many parameters had to be specified in order to complete assembly process in defined real-time. But, tuning those parameters through experimental discovering for improved performance was time consuming process.
The main contribution of this work is the use of a task replanning approach in combination with robot learning from experimental setup. We propose neural network based learning which gives us new successful vibration solutions for each stage of reducer. With this extended optimal vibration values as source information, we introduce Deterministic search strategy in scope of Robot Assembly Replanning Agent.

2. Machine learning

Machine learning usually refers to the changes in systems that perform tasks associate with artificial intelligence date. The changes might be either enhancement to already performing systems or synthesis of new system. A learning method is an algorithm (usually implemented in software) that estimates an unknown mapping between a systems input and outputs from the available data set. Learning is required when these mappings cannot be determined completely in advanced because of a priory uncertainty date (Farrell & Baker, 1993).
Generally speaking, there are two types of learning: supervised and unsupervised. These algorithms vary in their goals, in the available training data sets, in the learning strategies and representation of data.
Supervised learning requires a trainer, who supplies the input-output training instances. The learning system adapts its parameters by some algorithms to generate the desired output patterns from a given input pattern. In absence of trainers, the desired output for a given input instance is not known, and consequently the learner has to adapt its parameters autonomously. Such type of learning is termed unsupervised learning.

When the data are preprocessed and when we know what kind of learning task is defined for our application, it is important to make decision about the application of one or more of machine learning approaches. The most frequently used techniques include statistical methods (involve Bayesian inference), symbolic, inductive learning algorithms (decision building tree), cluster analysis, multiple-layered, feed-forward neural networks such as Backpropagation networks, fuzzy logic and evolution-based genetic algorithms (Kantardzic, 2001). These techniques are robust in their ability to analyze user queries, identify users' information needs and suggest alternatives for search.

3. Robot learning

Over the last few years, a number of studies were reported concerning machine learning and how it has been applied to help robots to improve their operational capabilities. Typical "things" that are learnt by robots are "how" to perform various behaviors: obstacle avoidance, navigation problems, planning robot control, etc. Imitation learning has helped significantly to start learning with reasonable initial behaviour.

It is difficult to define a coherent experimental method for robot learning (Wyatt et al., 1999). That is partly because the robot's behaviour may be the product of the robot's learning algorithm, it's initial knowledge, some property of the it's sensors, limited training time, stochastic actions, real-time responses, online learning, the environment or of an interaction between some subset of these. All of this makes it very difficult to interpret results. The robot learning experiments must be designed so as to generate meaningful results in the face of such complexity.

Essentially, we can define the robot learning as one of learning a policy function π from some set of sensory states S to some set of actions A. In order words, a task-dependent control policy π maps a continuous-valued state vector $\mathbf{x}$ of a controlled system and its environment, possibly in a time t dependent way, to a continuous-valued control vector $\mathbf{u}$:

$$u = \pi(x, t, \theta) \tag{1}$$

The parameter vector θ contains the problem-specific parameters in the policy π that need to be adjusted by the learning system. Examples of policy functions include desired control behaviours for mobile robots, such as avoiding obstacles, following walls, moving a robot arm to pick up some object.

Approaches to robot learning can be classified using three dimensions: *direct versus indirect control*, *the used learning method* and *the class of tasks* in question (Schaal, Atkenson, 2010).

How the control policy is learned, can be proceed in many different ways. Assuming that the model equation (1) is unknown, one classical approach is to learn these models using methods of function approximation and then compute a controller based on the estimated model, which is often discussed as the certainty-equivalence principle in the adaptive control. Such techniques are summarized under the name *model-based learning*, or *indirect learning* or *internal model learning*. Alternatively, *model-free learning* of the policy is possible given an optimization or reward criterion, usually using methods from optimal control or reinforcement learning. Such model-free learning is also known as direct learning, since the policy is learned directly, i.e., without a detour through model identification.

From the viewpoint of machine learning, robot learning can be classified as *supervised learning, reinforcement learning, learning modularizations or learning feature representations that subserve learning.*

We can distinguish two supervised paradigms, inductive concept learning and explanation-based learning (Mahadevan, 1996). Inductive concept learning, assumes that a teacher presents examples of the target function for the robot. In this paradigm, the temporal credit assignment problem is non-existent, since the teacher is essentially telling the robot what action to perform, in some situation. In explanation-based learning, the teacher not only supplies the robot with example of the target function, but also provides a domain theory for determining the range of sensory situations over which the example action is useful. It can be a logical function or a neural network, or even an approximate qualitative physics based theory.

The unsupervised paradigms involve reinforcement learning and evolutionary learning. In *reinforcement learning*, the learner does not explicitly know the input-output instances, but it receives some form of feedback from its environment. The feedback signals help the learner to decide whether its action on the environment is rewarding or punishable. The learner thus adapts its parameters based on the states (rewarding/punishable) of its actions. Intuitively, RL is a process of trial and error, combined with learning. There are several popular methods of approaching model-free robot learning. Value function-based methods are discussed in the context of actor-critic methods, temporal difference (TD) learning and Q learning. A novel wave of algorithms avoids value functions and focuses on directly learning the policy, either with gradient methods or probability methods.

The evolutionary learning is very similar to reinforcement learning, in that the robot is only provided with a scalar feedback signal, but the differences is in term of learning (online vs. offline), etc.

It is useful too to distinguish between several general classes of motor tasks that could be the goal of learning. *Regulator tasks* keep the system at a particular set point of operation-a typical example is a balancing a pole on a finger tip or standing upright on two legs. *Tracking tasks* require the control system to follow a given desired trajectory within the abilities of the control system. *Discrete movement tasks*, also called one-shot tasks, are defined by achieving a particular goal at which the motor skill terminates (basketball foul shot). *Periodic movement tasks* are typical in domain of locomotion. *The complex movement tasks* are composed of sequencing and superimposing simpler motor skills, e.g. leading to complex manipulation skills like assembling a bookshelf etc.

In order to achieve faster and reliable above specified complex robot assembly process in this research, we validate the results concerning the robotic assembly by introducing of learning strategies. First, the supervised (neural network) based learning is capable to reproduce the training data and to form clutter of adjustable vibrations for assembly process. Second, the unsupervised form of learning is used to reach a goal matting point using minimal path searching actions. It is equipped with reinforcement signal detection, which can measure physical aspect of mating process (model-free learning). The robot moves with reward in case of tolerance compensation. In case of jamming, Robot Assembly Replanning Agent uses this signal as error detection in system and replanns actions in order to achieve a goal position.

4. Planning agents

Intelligent agents are able to perceive their environment and respond in a timely fashion to changes that occur in it in order to satisfy their design objectives (Wooldridge, 2008). They are able to exhibit goal-directed behaviour by taking the initiative in order to satisfy their design objectives. But for non-functional systems, the simple model of goal-directed programming is not acceptable, as it makes some important limiting assumptions. In particular, it assumes that the environment does change and if the assumptions underlying the procedure become false while the procedure is executing, then the behaviour of the procedure may not be defined and it will be crash. In such environment, blindly executing a procedure without regard is poor strategy. In such dynamic environments, an agent must be reactive, i.e. it must be responsive to events that occur in its environment.

Building purely goal-directed systems is not hard, but it is hard building a system that achieves balance goal-directed and reactive behaviour. The agents must achieve their goals systematically using complex procedure-like patterns of action.

We assume that the environment may be in any of a finite set E of discrete, instantaneous states:

$$E=\left\{e,e^{'},...\right\} \tag{2}$$

Agents are assumed to have a finite repertoire of possible actions available to them, which transform the state of the environment

$$A_c=\left\{a_0,a_1,...\right\} \tag{3}$$

A run r of the agent in an environment is thus a sequence of interleaved environment states and actions:

$$r: e_0 \xrightarrow{a_0} e_1 \xrightarrow{a_1} e_2 \xrightarrow{a_2} e_3 \xrightarrow{a_3} ... \xrightarrow{a_{n-1}} e_n \tag{4}$$

We model agents as functions which map runs to actions:

$$A_g : R^E \rightarrow A_C \tag{5}$$

where R^E is subset of these that end with environment state.

Means-ends reasoning is the process of deciding how to achieve an end using the available means (actions that can perform). *Means-ends reasoning is known as planning*.

A *planner* is system that takes as input the following: representation of a goal, the current state of the environment and the actions available to the agent. As output, a planning algorithm generate a plan P. A plan P is a sequence of actions:

$$P=\left\{a_1,...a_n\right\} \tag{6}$$

Many agents must have reactive role in order to achieve goal, i.e. agent must *replann*. In this case agent has next structure:

$$P^{'}=\left\{a_1,a_i^{'},a_{i+1}^{'}...a_n\right\} \tag{7}$$

In practical reasoning agents, the plan function is implemented by giving the agent a plan library. The plan library is a collection of plans, which an agent designer gives to an agent. The control cycle of decision-making process of agent is a loop, in which the agent

continually observes the world, decides what intention to achieve, uses means-ends reasoning to find a plan to achieve these intentions and execute the plan (replann).
Learning has an advantage that it allows the agents to initially operate in unknown environments and to become more competent than its initial knowledge alone might allow. The agent decides on actions based on the current environmental state and through feedback in terms of the desirability of the action (reward), learns from interaction with the environment.
Examples of reaching a desired goal, avoiding obstacles, self-collisions, etc. using a combination of robot learning and task replanning are presented in (Banjanović-Mehmedovic, et.al., 2008), (Ekvall & Kragic, 2008).

5. Robot assembly system

5.1 Assembly system

The main difficulty in assembly of planetary speed reducers is the installation of tube over planetary wheels. Namely, the teeth of all three planetary wheels must be mated with toothed tube. Fig. 1. presents a). only one stage of planetary reducer, and b). planetary speed reducer (cross-section 20mm, height five degrees 36mm), which has been used for experiments.

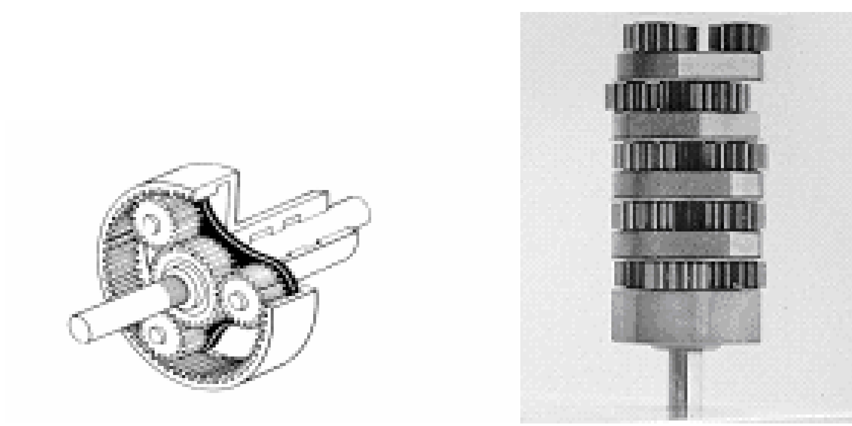


Fig. 1. One stage of planetary reducer, b). View inside of planetary speed reducer.

In this research has not been considered the complete assembly of each part of planetary reducer but only the process of connecting the toothed tube to five-stage planetary reducer. By solving the problem of assembly the gears, there will be no problem to realise complete assembly of planetary speed reducer.
For the process of assembly, the vertical-articulated robot with six-degrees of freedom, type S-420i of the firm FANUC has been used, completed by vibration module (Fig. 2.), developed at Fraunhofer- Institut für Produktionstechnik and Automatisierung (IPA) in Stuttgart, Germany. Total form of movement should be produced by vibration module to allow the fastest possible way of mating the tube with base part of planetary reducer respectively to compensate tolerance by vibration (Schweigert, 1995).
According to the functioning the individual systems of tolerance compensation can be divided into (Bernhart & Steck, 1992):

- controllable (active) system for tolerance compensation in which, on base of sensor information on tolerance, the correction of movement is made for the purpose of tolerance compensation

- uncontrollable (passive) system for tolerance compensation in which the orientation of external parts is achieved by the means of advanced determined strategy of searching or forced by connection forces
- combination of above two cases.

For this system of assembly (Banjanovic-Mehmedovic, 1999), the passive mechanism of tolerance compensation has been used with specially adjusted vibration of installation tools. The assembly process starts with gripe positioning together with toothed tube exactly 5mm above the base part of planetary reducer and than moving in direction of negative z-axis in order to start assembly (Fig. 2.).

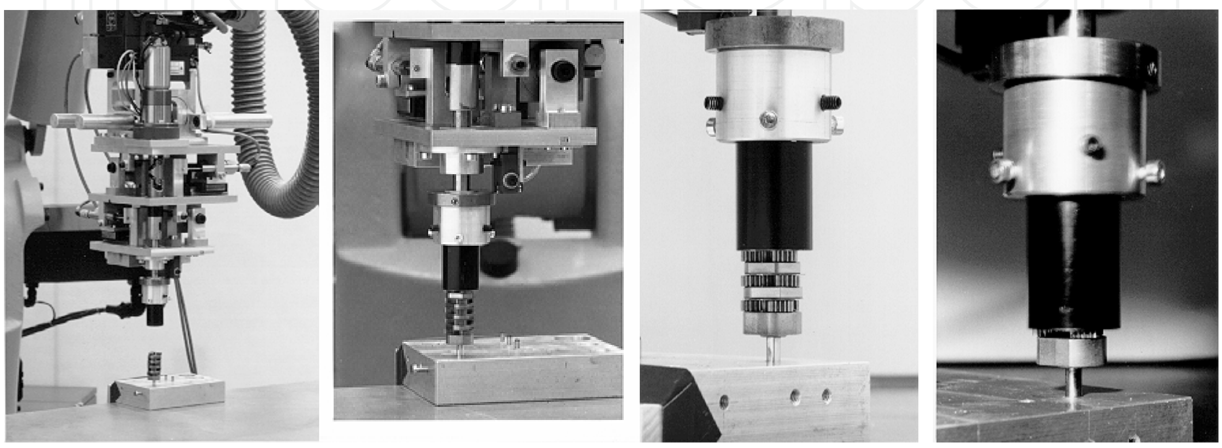


Fig. 2. Particular phases of assembly process.

The analysis of assembly process shows that movement based on vibration and rotation act positively on the course of process. Vibration module should be able to produce vibration in x- and y- direction, and rotation around the z-axis. Sensors (inductive sensor of passed way and vicinity) necessary in process of assembly ware mounted on vibration module. There was a special controlling card developed for control by step-motor and magnets for generating vibrations on vibration module.

5.2 Search strategy

The complex systems are often modelled according to either state-based or an event-based paradigm. While in state-based model, the system is characterized by states and states changes, in the latter case is characterized by event (actions) that can be performed to move from one state to another (H.ter Beek et.al., 2008).

Transition system is described with quadruple (S, s_0, A_C, R), where S is set of states, s_0 is initial state, A are transition from one state to another and R is transition relation. In our research, we used this concept in order to describe the relationships between the parts being assembled. Namely, the states are assembly parameters–vibration amplitudes and frequencies for each planetary reducer stage and transition action are used to move through assembly process from one stage to another of planetary reducer.

During the robot assembly of two or more parts we encounter the problem of tolerance compensation. For automatic assembly the tolerance is especially difficult problem because in process of mating it must be compensated but it takes time and requires corresponding algorithms.

In order to compensate tolerance during robot assembly, we use the *'search strategy'*, which adjusted amplitudes and frequencies to optimal values gained from experimental experience (amplitude of upper plate, amplitude of down plate, frequency of upper plate, frequency of down plate) (Fig. 3.). In case of jamming from different physical reasons (position, friction, force etc.), robot returned to beginning of current reducer stage, where the jamming was made. The search strategy tried three times to continue assembly process with another optimal assembly vibration parameter stage set values. It exploited the technique of blind search in optimal parameter space with repeated trials at manipulation tasks. When the jamming has been overcome, robot kept moving until it reached the final point in assembly. On the opposite, flashing of red lamp informed the personnel that there has been a jamming.

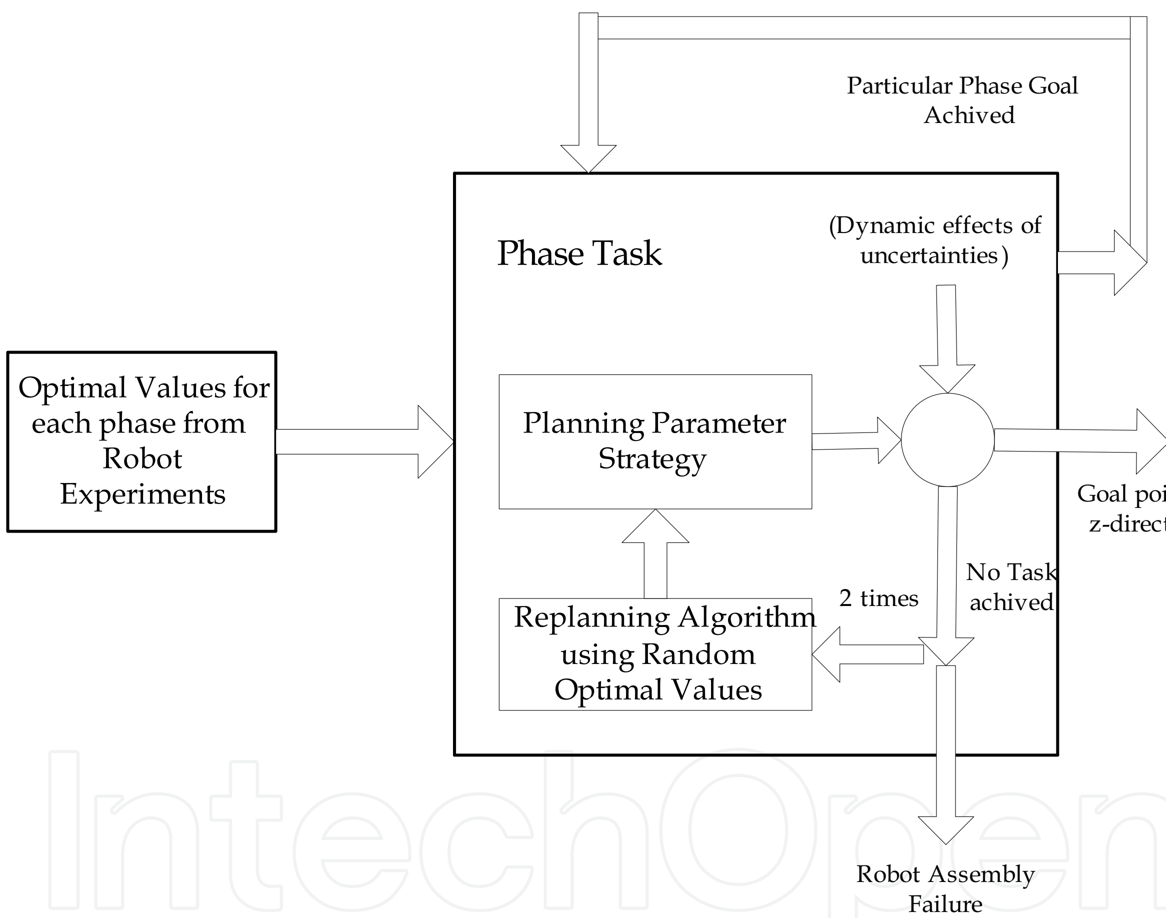


Fig. 3. Search strategy in experimental robot assembly.

There were extensive experimental complex investigations made for the purpose of finding the optimum solution, because many parameters had to be specified in order to complete assembly process in defined real-time. But, tuning those parameters through experimental discovering for improved performance is time consuming process.

The search strategy involved in assembly experiments exploited the technique of blind search of optimal vibration values in repeated trials in each stage. If selected optimal value is in discontinuity area, then the path between one selected optimal stage parameter set and another will be outside of cone (Fig. 4.).

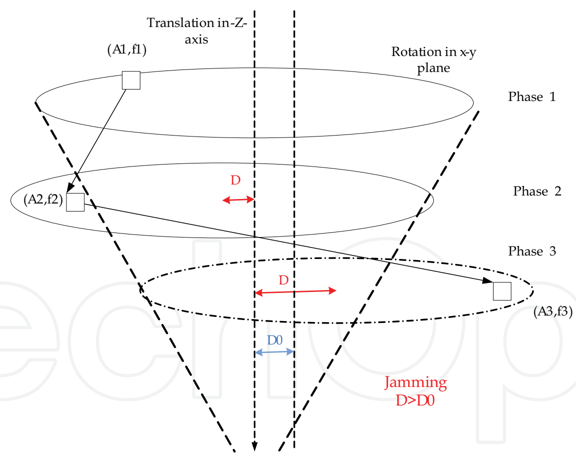


Fig. 4. Transition problems between states inside Search Strategy.

In this case, the tolerance compensation isn't achieved, because position tolerance of some stage D is greater than admitted position tolerance D0. What is solution for this? In order the path between two phases would be in cone towards stable tolerance compensation, we need *deterministic transition action* (directed path between vibration states based on minimal path finding).
To make this search strategy more intelligent, additional learning software was created to enable improvements of performance.

6. Robot assembly replanning agent

Today robot need to react to stochastic and dynamic environments, i.e., they need to learn how to optimally adapt to uncertainty and unforeseen changes (Schaal&Atkenson, 2010). The robot learning covers a rather large field, from learning to perceive, to plan, to make decisions etc.

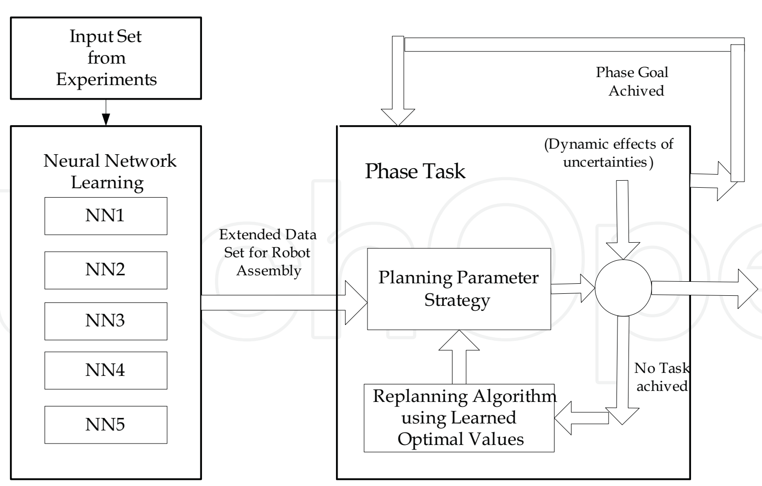


Fig. 5. Robot Assembly Replanning Agent.

Learning control is concerned with learning control in simulated or actual physical robots. It refers to the process of acquiring a control strategy for a particular control system and particular task by trial and error.
Task planning is the problem of finding a sequence of actions to reach a desired goal state. This is a classical AI problem that is commonly formalized using a suitable language to

represent task relevant actions, states and constraints (Ekvall & Kragic, 2008). The robot has to be able to plan the demonstrated task before executing it if the state of the environment has changed after the demonstration took place. The objects to be manipulated are not necessarily at the same positions as during the demonstration and thus the robot may be facing a particular starting configuration it has never seen before.
In this paper, we present a learning method in combination with robot path planning/replanning agent system. The performance of this method is demonstrated on a simulated robot assembly through intelligent agent system (Fig. 5.). We propose neural network based learning which gives us new successful vibration solutions for each stage of reducer. With this extended vibration parameters as source information for Planning/Replanning Task, we introduce advanced search strategy of robot assembly.
In the replanning scheme, the error model is used to model various dynamic effects of uncertainties and physical constraints by jamming. Combing the efforts of the planner and learned optimal values, the replanner is expected to guarantee that agent system enters the region of convergence of its final target location.

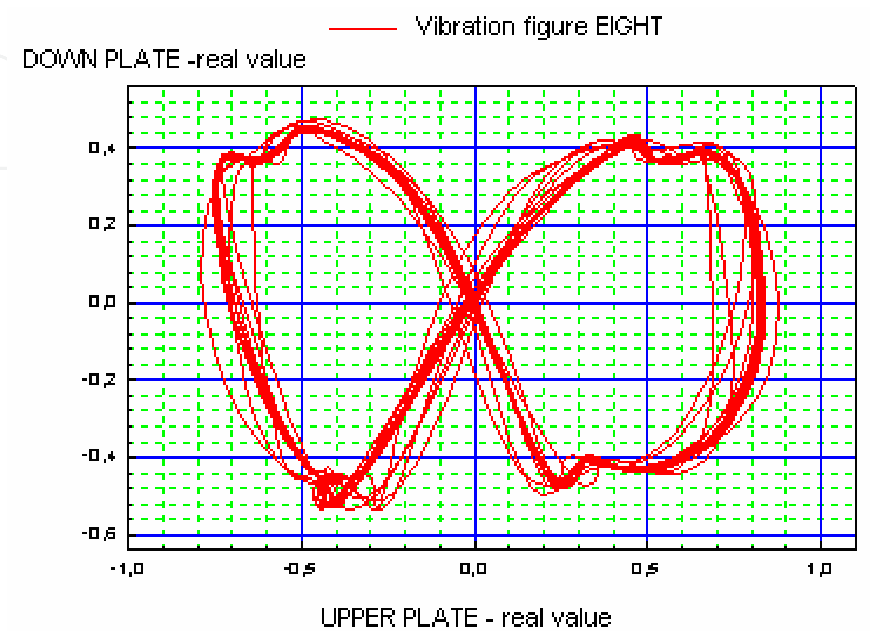
6.1 Neural network based vibration parameters learning

The artificial neural networks (ANN), with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze. This expert can then be used to provide projections given new situations of interest and answer to question “what if” (Stergiou & Siganos, 1996). Another reason that justifies the use of ANN technology, is the ability of ANNs to provide fusion of different information in order to learn complex relationships among the individual values, which would otherwise be lost if the values were individually analyzed.
There exist many types of neural networks, but the basic principles are very similar. Each neuron in the network is able to receive input signals, to process them and to send an output signal. The neural network has the power of a universal approximator, i.e., it can realize an arbitrary mapping of one vector space onto another vector space. The main advantage of neural networks is that they are able to use some a priori unknown information hidden in data, but they aren’t able to extract it. Process of ‘capturing’ the unknown information is called ‘learning of neural network’ or ‘training of neural network’. In mathematical formalism to learn means to adjust the free parameters (synaptic weight coefficients and bias levels) in such a way that some conditions are fulfilled (Svozil et al., 1997).
Neural network based learning is used in this research to generate wider scope of parameters in order to improve the robot behaviour. The parameter vector θ contains the problem-specific parameters in the policy π that need to be adjusted by the learning system. The amplitude and frequencies vibration data is collected during assembly experiments and is used as sources of information for the learning algorithm.

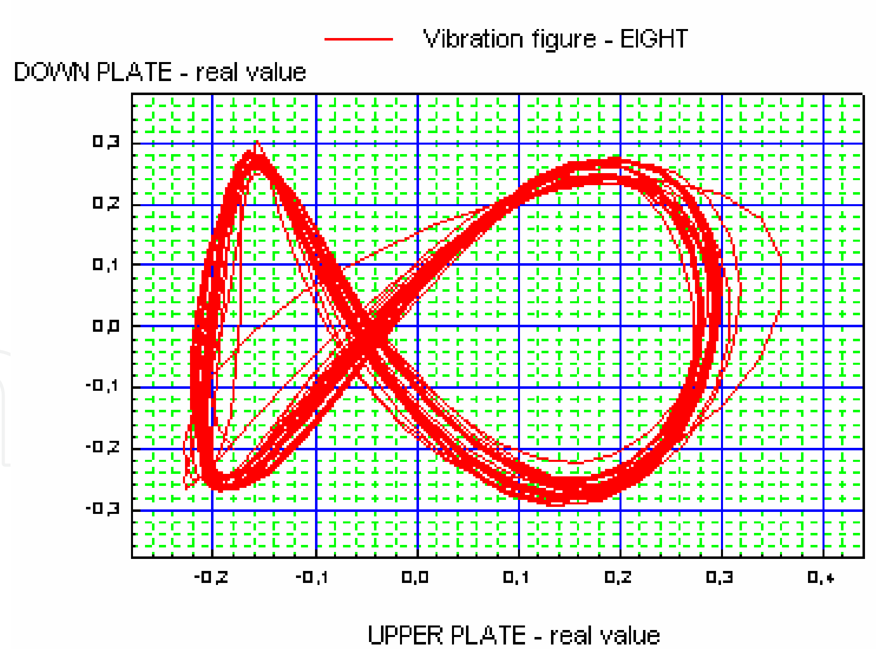
$$u = \pi\left(x, t, A, f_r\right) \tag{8}$$

By starting the robot work, vibration module vibrated with determined amplitude (to +/- 2mm) and frequency (to max. 10Hz) for each stage of reducer. For those experiments, the vibration figure horizontal EIGHT (Fig. 6) is used (the frequency ratio between down and above plate is f_D/f_U=2).

As optimum values of amplitudes of down and above plate that were valid for all stages of reducer are $A_D=A_U=0.8$mm. From experiments, we gained that smaller frequencies of vibration were better ($f_D/f_U=4/2$ or $6/3$) for 1-2 stage (counting of stages starts from up to down), while for each next stage the assembly process was made better with higher frequencies ($f_D/f_U=8/4$ or $10/5$).



a)



b)

Fig. 6. Vibration figure-EIGHT: a) (1-2 stage; $f_D/f_U=4/2$ $A_D/A_U=1.4/1.4$); b) (3-4 stage; $f_D/f_U=10/5$ $A_D/A_U=0.5/0.5$).

Multi-layer feed-forward neural networks (MLF), trained with a back-propagation learning algorithm, are the most popular neural networks. In our research we used MLF neural network contains 10 tansig neurons in hidden layer and 1 purelin neuron in its output layer. The feed-forward neural networks were formed and tested for each stage of assembly process. Each one was initialized with random amplitudes A_U=A_D=A_i between 0 and 2 and frequencies values f_i between 0 through 4. Namely, the range of the frequencies measurement is normalized by mapping from frequencies ratio f_D/f_U=(4/2, 6/3, 8/4,10/5) onto the range of the state frequencies values (0 through 4). To training the MLF network, we used 35 vibrations sets for each 5 phases of assembly. The mean square errors (MSE) during the training of 5 MLF networks were achieved for 7-10 epochs. Two thousand data points were taken as a testing sample.

The following picture (Fig. 7.) presents network's trying to learn the new optimal stage vibration sets indicated by their respective picture. Each frame consists of the network's training true regions (circles mark) and network's training bad regions (rectangle marks).

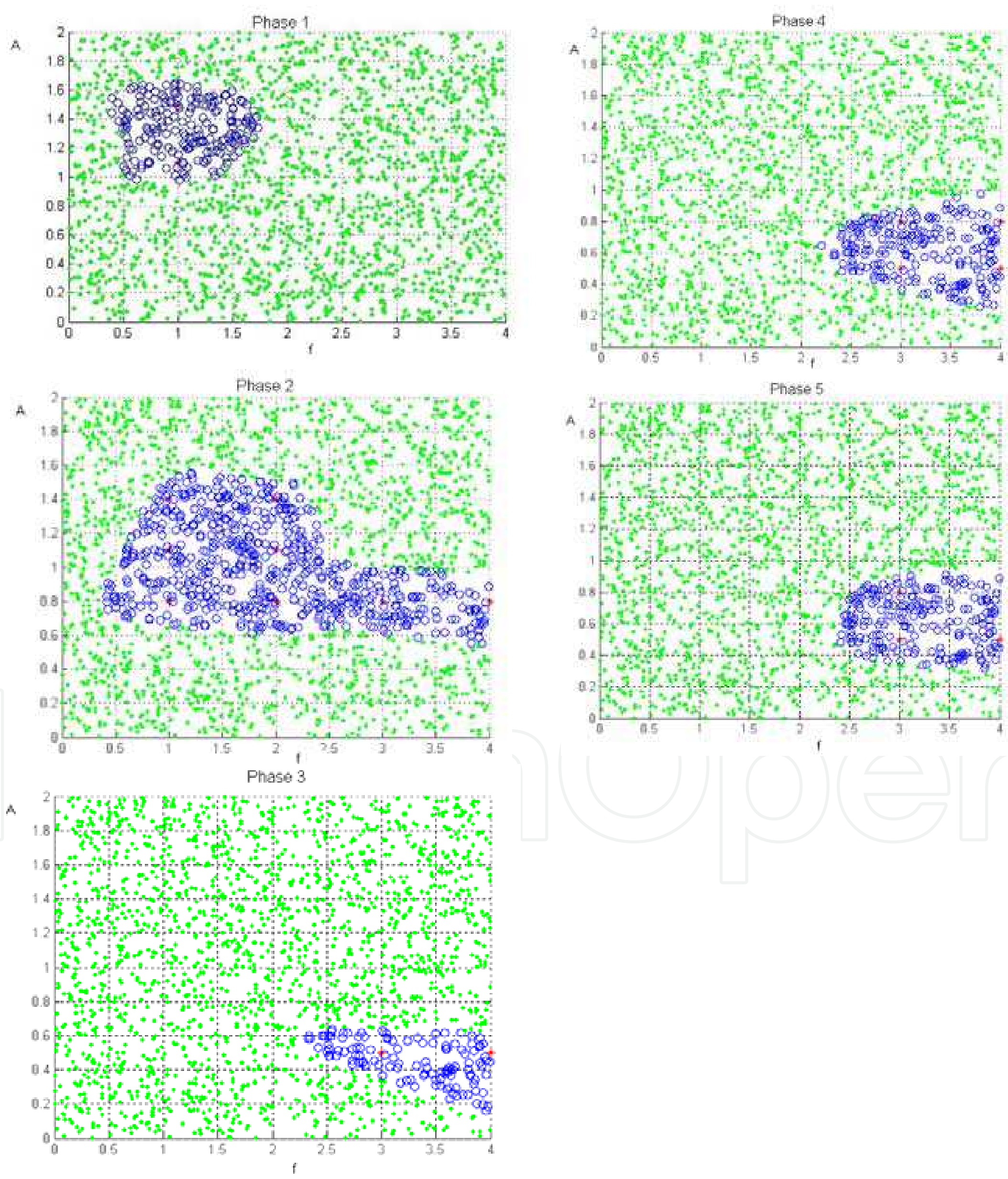


Fig. 7. Results of neural network training for all 5 stages

The results show that the scope of adjusted vibration parameters obtained from autonomous learning is extended in respect to adjusted vibration sets from experimental robot assembly. We can see that critical moment in assembly process is second phase, which presents medium clutter position of optimal vibration parameter sets through stages. Phases 2 presents discontinuity between first and third phase in clutter space. It can be reason for advanced form of planning/replanning too.

6.2 Advanced replanning strategy

The problem with applied search strategy in experiments was in case of behaviour switching (case of assembly jamming). The search strategy tried to continue assembly process with another optimal, but blind chosen parameter state value. With updated search strategy, named *Deterministic search strategy*, we propose next paradigm:

1. In order to have *deterministic transition action (DTA),* minimal distance is used between vibration state sets. DTA finds *minimal distance vector* from selected optimal value (A(i),f(i)), i=1,..N from current extended vibration state s(k) gained from learning process towards next vibration state s(k+1).

$$V_{path}(k) = \min\left(\left(A_o(k), f_o(k)\right) - \left(A_i(k+1), f_i(k+1)\right)\right), k = 1,..4 \quad (9)$$

The minimal path between two phase is in cone and we have compensated tolerance (D<D0), see Fig. 8.

2. In case of jamming (in our simulator: *error event signal*), we propose *Replanning Algorithm with Learned Optimal values*, which offers new plan for path tracking during simulation of robot assembly. Fig. 8. presents next situation: system detect error event during second state of assembly and strategy try to continue assembly process with another optimal set value (A2′,f2′) from state s(2). This another value is optimal parameter value, with mean value of distance from state s(1) to state s(2). We make enough offset from this critical optimal point to another optimal solution. After that, strategy establishes action between values (A2′, f2′) and (A3′, f3′).

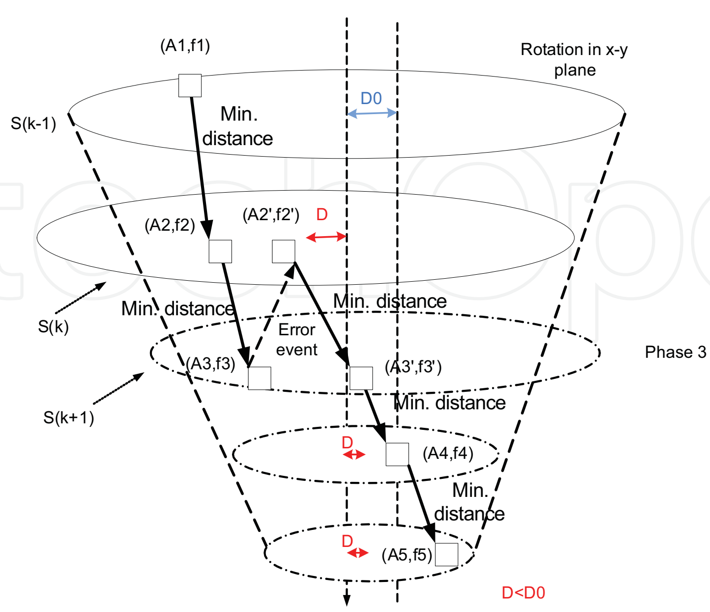


Fig. 8. Deterministic search strategy uses minimization of transition path between states and recovery parameter algorithm in case of jamming.

To demonstrate the validity of this paradigm, we present test results obtained by implementation of Robot Assembly Replanning Agent in Matlab. We use random start point in vibration parameter space (1.0,1.0), but system detects error event signal and tries assembly with new start vibration value (1.53, 1.27) (Fig. 9.).

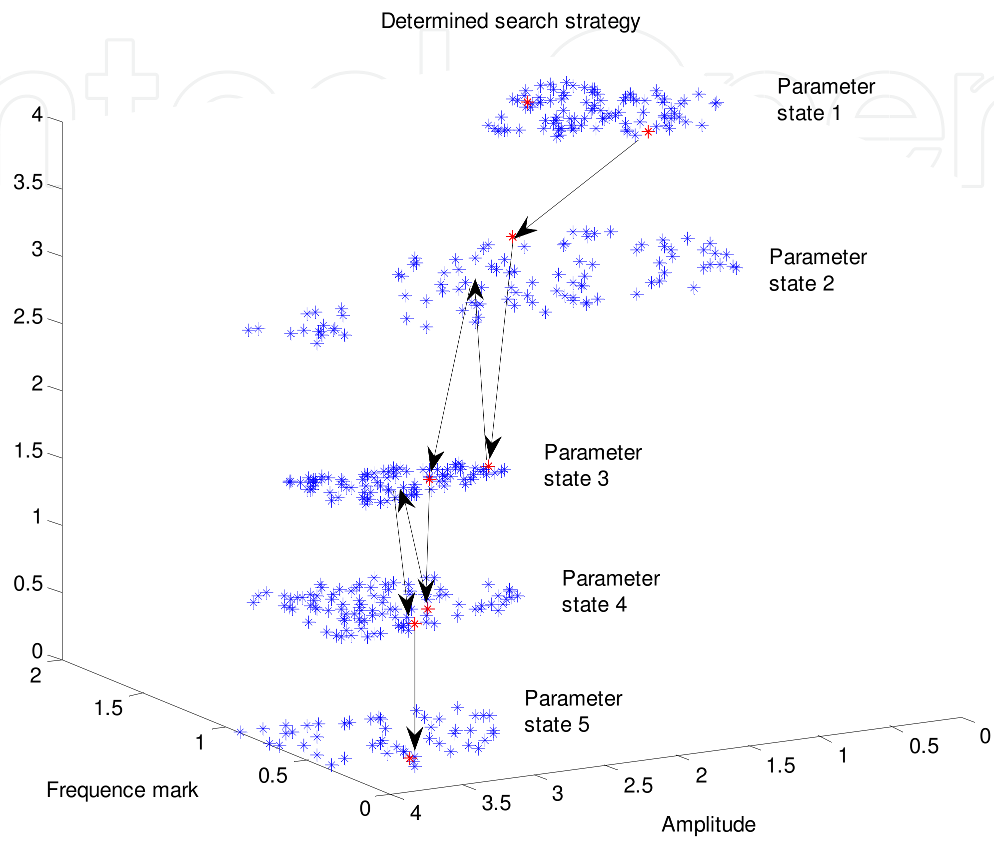


Fig. 9. Presentation of advanced search strategy in case of detecting error event signals.

In case of detecting of error event signal in second state, deterministic search strategy tries instead optimal value (0.52,2.72) to continue assembly process with another optimal assembly vibration parameter stage set value (0.49, 3.19). New transition action is made from this new optimal value from current state with minimal path distance towards optimal vibration parameter stage set in next state. But here, system detects new error event and tries assembly instead (0.52,3.14) with (0.36,3.42), until it reaches the final point in assembly simulation process.

7. Conclusion

There is enough space for investigation in this class of robot assembly search strategy, because the selection of assembly strategy is based on inspiration from human strategies. As an example of robot assembly, it was researched the complex assembly of toothed tube over planetary gears. Important contribution of paper is combination replanning task approach with learning approach in order to accommodate the uncertainty in complex assembly of tube over planetary gears. Two form of learning are proposed in state and action domain.

First, supervised neural network based learning is used to generate wider scope of state parameters in order to improve the robot behaviour. Second, the unsupervised learning is used to reach a goal matting point. Using Deterministic search strategy based on minimal path tracking as transition action between vibration states and replanning of actions in case of error signal detection in system, it is possible to involve intelligent control of robot assembly. Simulations were performed in domain of robot assembly to demonstrate usefulness of the presented method. Robotic provides an excellent test-bench for studying different techniques of computational intelligence.
Recent trends in robot learning are to use trajectory-based optimal control techniques and reinforcement learning to scale complex robotic systems. Future work in domain of replanning agent is research with genetic based replanning agent in order to accelerate the optimization speed of path planning technique.

8. References

Banjanovic-Mehmedovic, L. (1999). Robot Assembly of planetary motor speed reducer, *Proceedings of the VIII IEEE Electrotechnical and Computer Science Conference ERK '99;* Portoroz, Slowenia, pp.267-270

Banjanovic-Mehmedovic, L.; Karic, S.; Jasak, Z. (2008), Learning of Robotic Assembly based on Specially Adjustable Vibrations Parameters, *IEEE Symposium on Signal Processing and Information Technology (ISSPIT),pp. 123-128,* ISBN 978-1-4244-3555-5

Bernhart, W. & Steck, W. (1992). Vibration macht's möglich. *Technische Rundschau*, Vol.16, (1992) pp. 48-51

Bohman, J. (1994). *Mathematische Algorithmen für Fügeprozesse mit minimalen Rektionskräften und ihre Umsetzung in der Robotertechnik,* Doktor-Dissertation, Fakultät für Technische Wissenschaften der Technischen Universität Ilmenau

Brignone, L.; Sivayoganathan, K.; Balendran, V. & Horwarth, M. (2001). Contact localisation: a novel approach to intelligent robotic assembly, *Proceedings of International Joint Conference of Neural Networks*, pp. 2182-2187

Chan, S.P. (1995). A neural network compensator for uncertainties in robotic assembly. Journal of Intelligent and Robotic Systems, Vol 13., No.2, (June, 1995) pp. 127-141

Ekvall, S.; Kragic, D. (2008). Robot learning from demonstration: A Task-level Planning Approach, *International Journal of Advanced Robotic Systems*, Vol.5, No.3 (2008), pp. 223-234, ISSN 1729-8806

Farrell, J. & Baker, W. (1993). Learning Control Systems, In: *Introduction to Intelligent and autonomous Control,* Antsaklis, P. J. & Passino, K. M), (Ed.), Kluwer Academic Publisher

Kantardzic, M. (2001). *Data Mining, Concepts, Models, methods and Algorithms,* A John Wiley & Sons, Inc. Publication

H. ter Beek, M.; Fantechi, A.; Gnesi, S.; Mazzanti, F. (2008). *An Action/State-Based Model-Checking Approach for the Analysis of Communication Protocols for Service-Oriented Applications*, In: Lecture Notes in Computer Science, Vol. 4916/2008, pp. 133-148, DOI:10.11007/978-3-540-79707-4_11

Mahadevan, S. (1996). Machine learning for Robots: A Comparison of Different Pardigms, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-96)*, Japan

Miyazaki, F.; Ide, K.; Masutani, Y.; Ahn, D.S. (1993). Learning of robotic assembly based on force information, In: Experimental Robotics II, Lecture Notes in Control and Information Sciences, Volume 190/193, pp. (78-88), Publisher Springer Berlin/Heidelberg

Newman, W.S.; Branicky, M. & Pao, J.-H. (2001). Intelligent Strategies for Compliant Robotic Assembly, *Proceedings of 11th Yale Workshop on Adaptive and Learning Systems*, pp. 139-146, Yale

Schaal, S.; Atkenson, C.A. (2010) Learning Control in Robotics, *IEEE Robotics and Automation Magazine*, Vol.17, No.2, June 2010, pp. 20-29, ISSN 1070-9932

Siegwart, U. (1995). Taktile Präzisionsmontage mit Industrierobotern in der Feinwerktechnik. *wt-Produktion und Menagment 85,* (1995) pp. 33-36

Svozil, D.; Kvasnička, J. & Pospichal, J. (1997). Introduction to multi-layer feed-forward neural networks. *Elsevier Chemometrics and Intelligent Systems 39,* (1996), pp. (43-62).

Stergiou, C. & Siganos, D. (1996). Neural Networks. *SURPRISE 96*, Vol.4

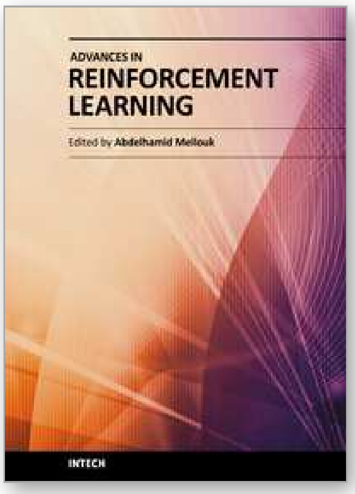
Vaaler, E.G. (1991). *A machine Learning Based Logic Branching Algorithm for Automated Assembly,* PhD, MIT

Wai, J. (2001). *Intelligent Robotic Learning using Guided Evolutionary Simulated Annealing,* MSc. Thesis, Case Western Reserve University

Wyatt, J.; Hoar, J.; Hayes, G. (1999). *Experimantal methods for robot learning*, In: Towards Intelligent Mobile Robots - Scientific Methods in Mobile Robotics, Nehmzow U.; Recce, M. & Bisset, D., (Ed.), Department of Computer Science

Xiao, J.; Zhang, L. (1995). A Geometric Simulator SimRep for Testing the Replanning Approach toward Assembly Motions in the Presence of Uncertainties, *Proceedings of the 1995 IEEE International Symposium on Assembly and Task Planning (ISATP'95),* pp.171-177

Wooldridge, M. (2008). *An Introduction to MultiAgent Systems*, John Wiley & Sons , Inc. Publication, ISBN 978-0-471-49691-5



Advances in Reinforcement Learning

Edited by Prof. Abdelhamid Mellouk

ISBN 978-953-307-369-9

Hard cover, 470 pages

Publisher InTech

Published online 14, January, 2011

Published in print edition January, 2011

Reinforcement Learning (RL) is a very dynamic area in terms of theory and application. This book brings together many different aspects of the current research on several fields associated to RL which has been growing rapidly, producing a wide variety of learning algorithms for different applications. Based on 24 Chapters, it covers a very broad variety of topics in RL and their application in autonomous systems. A set of chapters in this book provide a general overview of RL while other chapters focus mostly on the applications of RL paradigms: Game Theory, Multi-Agent Theory, Robotic, Networking Technologies, Vehicular Navigation, Medicine and Industrial Logistic.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821