



From Human–Machine-Interaction Modeling to New Concepts Constructing Autonomous Systems: A Phenomenological Engineering-Oriented Approach

D. SÖFFKER

Dynamics and Control, University of Duisburg, Germany; e-mail: soeffker@uni-duisburg.de

Abstract. Modeling of the Human–Machine-Interaction (HMI) gives ideas to transfer the developed modeling approach of human interaction with formalizable environments to technical systems, like mobile robots, to give them some kind of autonomy for interaction tasks. The important task of the proposed new kind of intelligent control is to respond autonomously and problem-equivalent to complex or unknown situations. This includes learning capabilities and a well organized dynamic memory. The contribution introduces into the developed Situation-Operator model as the texturing theoretical background, into the complete concept, and classifies the known control approaches in the developed system-theoretic view of control and interaction, with or without autonomy.

Key words: situation, operator, qualitative modeling, machine learning, autonomous system, interaction.

1. Introduction

Whereas classic control schemes typically address issues relating to speed, accuracy and other (low-level) problems typically to physical oriented technical tasks, more complete theoretical models of system control or interaction behavior often are quite complex and unwieldy in unknown environments or situations.

In the sixties and seventies the human-control behavior was examined for stimulus-response tasks, e.f. describing the time behavior of human driving etc., e.g. (Schweitzer, 1970). In the nineties the Human–Machine-Interaction itself has been focused more intensively. An actual overview to the developed approaches is given by Cacciabue (1998). Different research directions have been established, which are oriented between Artificial Intelligence (AI) approaches and phenomenological macro cognition oriented engineering approaches (Cacciabue, 1998). In (Söffker, 1999; 2001) a modeling approach of human interaction with formalizable technical environments is developed. Core of the work is a specified Situation-Operator model (SOM). In contrast to known procedures (Görz, 1995; Müller, 1993; Sandewall, 1994) the developed approach is neither based on exact temporal logic nor assumes a mathematical perfect understanding of context structures.

Gigerenzer and Goldstein (1996) show that unsatisfying classical norms of rational inference fast and frugal algorithms can lead to effective rationality. This includes the possibility working with non-perfect algorithms to imitate aspects of human learning. The idea is to transfer the engineering oriented approach (Söffker, 2001) to technical systems to give them some kind of memory and internal organizing features for autonomous interaction capabilities. In the sequel the developed SOM-technique is used to describe the 'human controller' (HC) as well as an autonomous system (AS) and will be summarized as intelligent system (IS).

2. From Scenes and Actions to Situations and Operators

Core of the approach is the assumption that changes of the considered parts of the Real World (RW) are understood as a sequence of effects described by the items scenes and actions (Söffker, 1999; 2001). In the proposed approach the definition of the items scene and action are coordinated in a double win. They are related to each other and they relate the assumed structure of the RW to the structure of the database – called mental model – of IS.

ISs are included in the Real World (RW). While ISs are interacting with the RW, they can change the RW. Depending on their principal sensor inputs, their natural (HC) or technical (AS) perceptions, and on the related knowledge base, the ISs adapt and learn only parts or aspects of the RW. These parts are modeled using the developed situation and operator calculus. The describable part of RW is called a system.

The item situation, which is (in contrast to (McCarthy and Hayes, 1969)) a time-fixed, system-, and problem equivalent one, is used describing the system structure (as a part of the RW). Here only the logical structure of the 3D-space, time and functional-oriented connections are of interest. The item operator is used to model effects/actions changing scenes (modeled as situations) in time. The situation S consists of characteristics C and a set of relations R . The characteristics are linguistic terms describing facts (as qualities). This will include physical, informational, functional, and logical connections. To describe the relations r_i known problem related modeling techniques, like ODEs, DAEs, Algorithms or even graphical illustrations (e.g. Petri-nets) can be used.

The SOM-approach only gives the frame to model the structure of the relations, and therefore maps the 'reality' into a structural framework. This is useful describing problems, where the system structure is complex and cannot be modeled using single approaches. This is especially useful to describe interactions between IS and its environment.

The introduced item characteristic (C) also includes the possibility of representing time-dependent parameters P . The set of relations R (of C s) fixes the structure of the considered scene of the world modeled as situation S . The introduced situation concept consequently allows the integration of different types of engineering-like descriptions. As an example in Figure 1 the complex physical

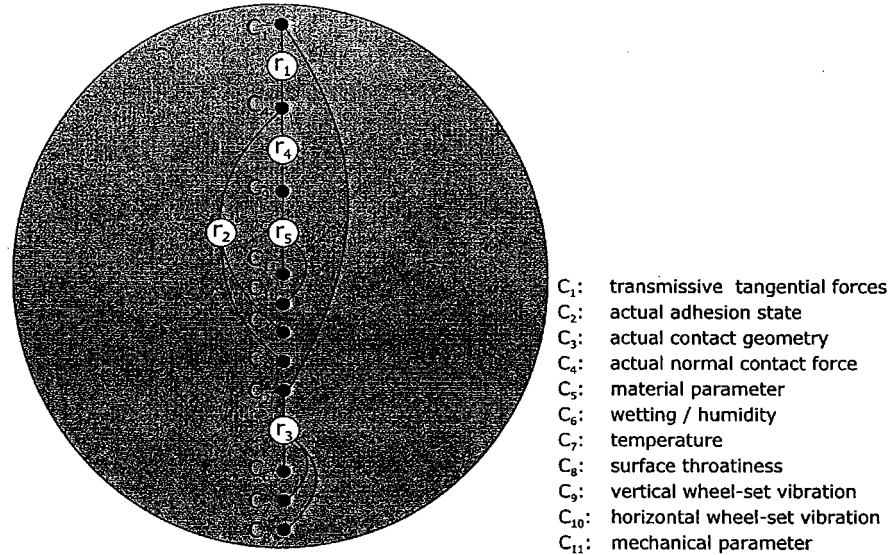


Figure 1. Structure of the proposed item situation S (Detailed example: Rail-wheel contact modeling).

situation of the rail-wheel physics and the dynamics interaction and the structure of the related modeling is given. The illustration shows the dependencies of the considered mapping from the real world problem to the engineering oriented modeling using ODEs and case-depending algorithms into a qualitatively modeled and graphical illustrated network.

For this example shown in Figure 1 the relation r_1 connects C_1 and C_4 , which represents the tangential contact force T_ξ (C_1) and the normal force N (C_4). The function is modeled by

$$T_\xi = \alpha T_\xi^{\text{lin.}} = -\alpha abGC_{11}\eta, \quad (1)$$

with

$$\alpha = 1 - \frac{1}{3} \left(\frac{T_\xi^{\text{lin.}}}{\mu N} \right) + \frac{1}{27} \left(\frac{T_\xi^{\text{lin.}}}{\mu N} \right)^2 \quad \text{for } T_\xi^{\text{lin.}} \geq 3\mu N, \quad (2)$$

$$\alpha = \frac{\mu N}{T_\xi^{\text{lin.}}} \quad \text{for } T_\xi^{\text{lin.}} < 3\mu N, \quad (3)$$

with coefficients and variables C_{11} , a , b , G , η , μ , which are not declared here, due to principal aspects of using relations r_i . Details of the mentioned problem can be found in (Söffker, 1999).

The illustrated item operator is used for the modeling of (a) internal (passive) connections of situations (cf. Figure 1) and (b) changes between situations.

The operator O (cf. Figure 2) is understood and modeled from a functional point of view: the operator is an information-theoretic term which is defined by

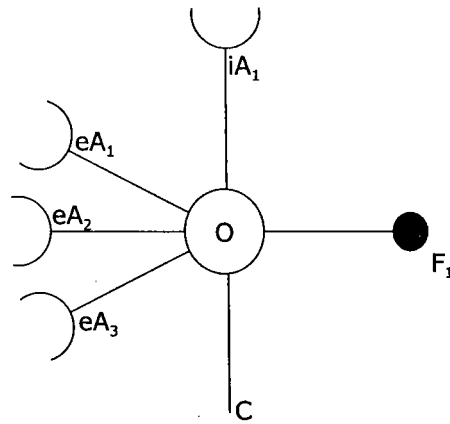


Figure 2. Structure of the proposed item operator O (principal example).

his function F (as the output) and the related necessary assumptions. Here explicit and implicit assumptions eA , iA are distinguished. F will only be realized, if the explicit assumptions eA are fulfilled. The iA includes the constraints between eA and F of the operator. The eA are of the same quality as the characteristics C of S . For the internal structure of the operator other descriptions like textual, logical, mathematical or problem-related descriptions are allowed.

The description of systems using a Situation-Operator model allows

- the mixture of different types of variable quantities,
- the integration of logical and numerical quantities, and
- the description of real-world problems using a mixture of a complex set of descriptions (variables), which is in interaction with IS.

Operators are used to model the system changes (changes of situations). This also defines the discrete events of the change of situations. Operators and situations are strongly connected due to the identity (partly or complete) of the characteristics of the situations and the explicit assumptions of the operators. This includes that the situation consists of ‘passive’ operators (internal causal relation: ‘because’), whereby the change is done by ‘active’ operators (external causal relation: ‘to’), shown in Figure 3. The change of the considered world results as a sequence of actions modeled by operators as illustrated in Figure 4.

Please note that operators correspond to situations. Both are not only used for structural organization of the system, but also for internal representation and storage of IS. They are the core/background of all higher organized internal operations and functions of the IS like learning, planning etc. (Söffker, 2001).

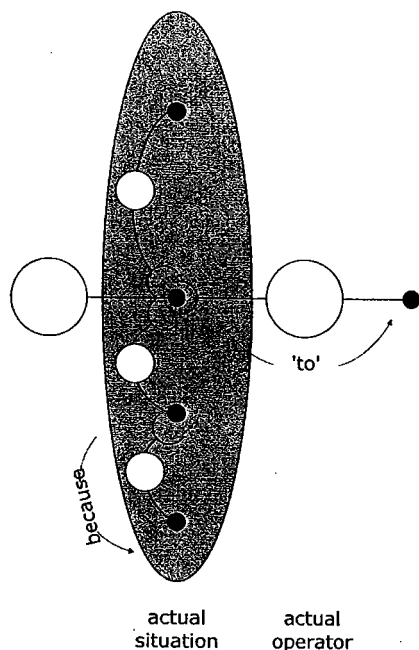


Figure 3. Connection between situation and operator.

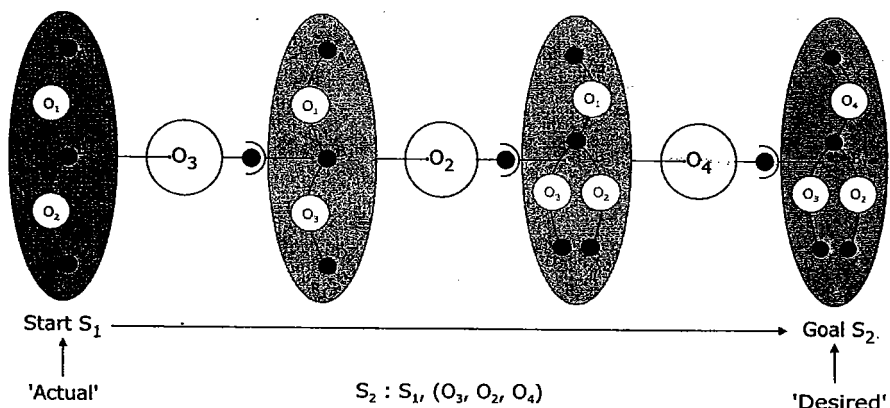


Figure 4. Sequence of operators changing the situations from the originator to the desired goal (example).

3. Learning

The following assumptions have been made:

- The problem-dependent structures of the real world scenes can be clearly identified as situation dependent R 's and C 's.
- The resulting identified S describes the RW in the way, that the relevant structure of the scenes and those of the identified S is equal.
- Operators are defined as time-independent.

Based on the introduced assumptions, learning appears as the definition/redefinition of operators, driven by the interaction between IS and the considered system, whereby the interaction can be intended or not, useful or not, and planned or not. This includes several different cases, where S_i denotes the i th situation, R_i denotes the i th set of relations and A_i, B_i, D_i denote a set of characteristics C_i describing the system structure of the i th situation.

A straight forward learning strategy includes the definition of O_1 by his induced (: active learning) or observed (: passive learning) situation changes,

$$O_i: S_i(R_i(C_i)) \rightarrow S_{i+1}(R_i(C_{i+1})), \quad R_i = R_{i+1}, C_i \neq C_{i+1}, \quad (4)$$

$$O_i: S_i(R_i(C_i)) \rightarrow S_{i+1}(R_{i+1}(C_i)), \quad R_i \neq R_{i+1}, C_i = C_{i+1}, \quad (5)$$

$$O_i: S_i(R_i(C_i)) \rightarrow S_{i+1}(R_{i+1}(C_{i+1})), \quad R_i \neq R_{i+1}, C_i \neq C_{i+1} \quad (6)$$

which includes possible changes of situation structures $R_i \rightarrow R_{i+1}$, characteristics $C_i \rightarrow C_{i+1}$ or both.

This forward learning and definition procedure of operators may be the main mechanism to map the outer world of the considered environment to the inner (mental) world of IS and to add new experiences into the 'database' of IS.

This assumes that IS is able to identify R_i, C_i from the available sensor inputs in combination with the actual knowledge. This can not be assumed in general. To overcome the included problems of learning coincidental coherencies and learning non-concrete coherencies due to insufficient memory – mental model (MM) – capabilities, it is necessary to include backward oriented learning abilities: this includes the ability to distinguish C s necessarily connected to R to those of coincidental presence and not directly connected to the problem structure.

EXAMPLE 1. The reality consists of $S_i(R_i(A_i, B_i), D_i)$ (R_i connects A_i and B_i , D_i is unconnected present) and the learning mechanism of IS assumes/identifies S_i as $S_i(R_i(A_i, B_i, D_i))$. After application of the O_1 -related action the system appears as $S_2(R_1(A_2, B_2, D_1))$, so O_1 can be defined by IS as

$$O_1: S_1(R_1(A_1, B_1, D_1)) \rightarrow S_2(R_1(A_2, B_2, D_1)). \quad (7)$$

Due to the contingencies of the reality it may happen that

$$S_1(R_1(A_1, B_1)), O_1 \rightarrow S_2(R_1(A_2, B_2)) \quad (8)$$

can be observed, which is in opposition to (7). So IS gets the chance to rebuild the O_1 definition by 'replaying' to find the true $S_1, O_1 \rightarrow S_2$ sequence redefining the operator O_1 . In the example this gives the opportunity to reject the coupling of D_1 to O_1 .

In this way learning appears as a strictly nonlinear procedure due to the strong connection of the definition process of operators to the actual context, which includes the individual initial conditions of IS (the actual S and MM).

EXAMPLE 2. The task of IS should be the realization

$$S_1(R_1(A_1, B_1, D_1)) \rightarrow S_2(R_1(A_2, B_2, D_1)). \quad (9)$$

IS will take O_1 , as learned (Equation (7)). Different results are possible:

- D_1 appears as learned, so O_1 seems to be confirmed.
- D_1 changes unexpectedly to D_2 or disappears.

As a result the reality may be in contradiction to the MM, so occurring differences give good reasons to reflect and change the definitions (previous learning procedures). It depends on internal features of IS to rebuild the MM immediately, after additional experiences or after extensive hypothesis oriented tests of the definition of O_1 or not.

Please note that this definitions of learning are independent from external commendations, penalties or rewards. The learning capabilities are the key feature for successful acting in unknown situations.

From a system-theoretic point of view the key feature is the model-updating capability of IS.

4. Planning and Acting

In this context planning is assumed as the internal preparation of the action or the series of actions to change actual S_{act} to desired ones $S_{des.}$, cf. Figure 4. Modeling of planning based on the SOM-technique includes a MM as a set of previously learned definitions/operators and the ability to identify the given goal $S_{des.}$ and $S_{act.}$. The goal elaboration is not considered here. To elaborate goals (or part goals) detailed procedures (as algorithms) have to be developed.

Planning includes the elaboration of a sequential ordered set of suitable O_i to solve the task $S_{act.} \rightarrow S_{des.}$. Due to the definition of S and O this can be done by comparison of C_i, eA, F applying a backward or forward inference strategy. The solvability strongly depends on the actual content of MM. If this can not be solved exactly (different reasons possible), practical planning procedures are possible which use operators which do not exactly fulfill the requirements (full set of Cs), but requirements close to the desired perfect ones. This will lead to testing strategies, associative combinations (where internal similarities between the relation ea, F of the supposed unknown, but perfect O and the C of known O exist). In reality conflicts may exist between goals, part goals, necessary actions, reachable situations and unexpected effects of 'known' operators. This may be typical for human interactions but also will appear for AS. The collection of possible human errors and the related SOM-oriented representation shows that there exists a large variety of practical problems (Söffker, 2001). In general solvable conflicts can be solved using decision making strategies with given goals. Therefore the solving strategy is to transform the problem to a higher level, where a solution can be given using an algorithm etc.

This includes the development and evaluation of alternative paths (operator sequences), the choice of weighting factors etc. and also strongly depends on the MM. In the (low level) case of scalar expressions (as characteristics and relations modeled by ODE) conflicts can be expressed by mathematical expressions, which can be solved perfectly or can be optimized using weighting functions to find compromises related to given goal functions. In the assumed general case of considering IS with formalizable and changing environments this problems are not considered up to now. Therefore game theory gives hints to the way the problem solution can be structured and solutions can be found.

The execution of the mentally prepared sequence of operators as actions realizes the interaction of IS with RW. The interaction itself gives a variety of learning sequences: the result of each action of IS can be compared with the predicted one to optimize the internal MM etc.

5. The Resulting Hierarchy of System Control

Here only a brief introduction into the system-theoretic consideration of different control approaches developed in (Söffker, 2001) is given.

Feedback control is understood 'as the operation that, in the presence of disturbances, tends to reduce the difference between the output of the system and some reference input and that does so on the basis of this difference' (Ogata, 1990). The influence/effect of the values to be controlled to the input values is realized by the controller. Characteristics of automatic control are

- the closed loop by feeding back values to be controlled and
- the automatic response.

The design of controllers gives

- fix strategies/rules/relations/or just a set of gains as control coefficients which connect system output with system input and
- a system extension to optimize the system behavior.

Based on the SOM-structured view to system-IS-interaction a new and more general view to automatic control appears.

SOM-BASED DEFINITION. Automatic control defines the autonomous transfer from actual to desired situations, whereby the inner structure or the modification of the inner structure of the considered and through the situation-space described system is used.

The proposed definition includes the old one, solves the problem of integrating algorithms/soft-computing algorithms, includes the interaction of IS, and – much more important – gives the view to new applications due to a homogeneous and uniform system.

5.1. EXAMPLE 3: SOM-VIEW TO CLASSICAL CONTROL

A controlled continuous SDF system, where the input (B) – output (A) behavior can be described using an ordinary differential equation, appears in the SOM-structured context as a fix situation. The connections are as follows:

Characteristic C_i :	A:	(Scalar) system output with time variant values
	B:	(Scalar) system input with time variant values
	C:	(Scalar) reference value
Relation R_i :	r_1 :	ODE (B input, A output)
	r_2 :	Controller rule (A, C inputs, B output)
Operator:		No operator exists, due to the fixed control law structure, this includes that nothing more will be changed by the controller

The characteristics are scalar and fixed, the parameters of A, B, C are time variant, the r_i are fixed, the implementation of control connects two situations (without and with control) by the application of r_2 , after the implementation of the control structure there is – from a SOM-theoretic point of view – only the fixed situation including the closed loop (as a closed chain of relations). This includes a fixed situation space, only depending on the dynamic behavior determined by the system behavior (described by r_i) and the input, which can not be changed by this type of control.

5.2. EXAMPLE 4: SOM-VIEW TO ALGORITHMS

An algorithm appears as a fixed sequence of situations, which may depend on external values. The algorithm steps directly correspond to operators. In contrast to IS-capabilities like cognitive based human behavior the complete algorithm is fixed before his execution.

Characteristic C_i :	Data of the algorithm
Relation R_i :	Internal connections between C_i , given by the problem modeling implemented in data-objects
Operator O_i :	Execution procedures change the objects of the algorithm: the data; the sequence of operators is defined before the operation itself, the situation space is previously defined.

The algorithm controls the changing of the objects of the algorithm: the data. The integrated feedback mechanism is not necessarily numerically defined, fur-

thermore logical comparison is possible (repeat-until, while-do, for-do-algorithm). Additionally feedback elements are possible integrating external effects which are combined with actual data outputs. Especially the dynamic sequences represent a higher quality of feedback, not only restricted to the comparison of (mostly scalar) numerical values and variables.

Without external effects there are not any feedback elements. With external effects (external given parameters, decisions etc.) algorithms also control the change and the flow of data, goal-oriented. The reachable situation space is defined in advance, which includes that algorithms are not completely autonomous.

5.3. EXAMPLE 5 : SOM-VIEW TO IS

An intelligent system is able to interact goal-oriented with unknown or complex environments. In contrast to classical control laws and algorithms IS are able to learn relations between observed sets of characteristics and to modify them.

Characteristic C_i :	Facts of the environments partially corresponding to mental elements of ISs database
Relation R_i :	Internal connections between C_i , given by the structure of the considered system environment
Operator O_i :	Active: Item which corresponds to the change of the system Passive: Item which maps outer world relations to inner mental relations

The key features of IS are learning and database capabilities. The main differences are:

- The interaction with a given environment is neither restricted to known (simple) single characteristics (classical control) nor to a known system (classical control, algorithm).
- The IS-system behavior can be defined in the moment of interaction, which will give a large flexibility, whereby with classical control the structural system behavior is fixed in advance (by given rules (e.g. coefficients)), or predefined as with algorithms.
- IS are able to learn, this includes 'cognition' about the interaction and the observed behavior of the environment. It should be noted, that this strongly depends on the actual internal (mental) model and on the sensor input capabilities and will lead to a highly nonlinear process.

5.4. NEW CRITERIA TO DISTINGUISH DIFFERENT CONTROL AND INTERACTION MECHANISMS

To distinguish different levels of control new criteria are necessary, which are suggested here:

- *Quality of the closing feedback and reference*

The comparison between the reference (which represents the goal) and the control value can be done in different ways and with different qualities, called characteristics here.

EXAMPLE 6.1 (Classical Control). Physical differences (numerically expressed) between scalar characteristics are widely used, the comparison is directly forwarded to the controller.

EXAMPLE 6.2 (Algorithm). Logical and/or numerical differences control that the part of algorithms, which includes control structures (cf. while-do, repeat-until, for-do constructs).

EXAMPLE 6.3 (IS). Generalized logical characteristics or sets of characteristics are used defining reference (goal) and actual situation.

- *Connection between control criteria and control goal*

The previously mentioned quality of the closing feedback is connected to the goal of the feedback itself.

EXAMPLE 7.1 (Classical Control). The fixed control law is designed with a fixed reference situation as goal. The fixed goal situation defines the fixed operator which is applied to reach the goal. The goal is represented by the reference value.

EXAMPLE 7.2 (Algorithm). The fixed algorithm also represents a fixed goal, resp. the steps (sequence) the goal.

EXAMPLE 7.3 (IS). The goal of the interaction can be changed during the interaction, the goal is represented by an internal situation. The goal gives the reference to develop strategies, for planning purposes and to define actual acting.

- *Time behavior during control*

EXAMPLE 8.1 (Classical Control). Classical controllers typically work time continuously, time discretely, sometimes event discretely.

EXAMPLE 8.2 (Algorithm). Algorithms (and mainly programs) are working time discretely.

EXAMPLE 8.3 (IS). The interacting behavior of IS with their environment is connected strongly with the goal and the tasks and the resulting actions. It

can be continuous, time discrete, or event discrete, but in general it is action discrete.

– *Variability of control (law)*

The variability of control defines the possibility to change the feedback and therefore the ‘control law’.

EXAMPLE 9.1 (Classical Control). No variability exists, may be that the fixed laws can be adapted using (fixed) adaptation rules.

EXAMPLE 9.2 (Algorithm). Some kind of variability exist, but is related to a previously defined set of rules/steps.

EXAMPLE 9.3 (IS). The IS is able to vary goals and rules. This is the key feature of IS. The resulting question is how to guarantee stability or success.

– *Anticipation of the situation trajectory*

EXAMPLE 10.1 (Classical Control). The situation trajectory usually consists of a single situation and a fixed set of characteristics, whereby the parameters are changing.

EXAMPLE 10.2 (Algorithm). The situation trajectory can be complex, is usually not fixed, and is open in a given situation space.

EXAMPLE 10.3 (IS). The situation trajectory may be unknown.

6. The Architecture of a SOM-Based Autonomous System

The architecture of the proposed behavior- and memory-based open system is given in Figure 5. Here additionally modules of hypothesis testing, of data based reconfiguration and, very important, of object- and scene recognition, scene and phenomenological interpretation are given for the example of a mobile robot.

It will be clear that the system is able to get an impression of his environment, depending on his sensor inputs, the observable change of the environment and on his interaction sequence. The system is designed for unknown environment, which includes that firstly interactions are needed to build the MM.

The basic modules

- electromechanical system of the mobile robot,
- basic control (drives, collision avoidance), and
- vision system
- object recognition

should not be declared here.

The function of the new modules for scene interpretation, phenomenological interpretation, planning, testing, and learning is declared briefly.

Scene Interpretation. The scene interpretation module connects the vision system/object recognition with the logical interpretation module. As input reference

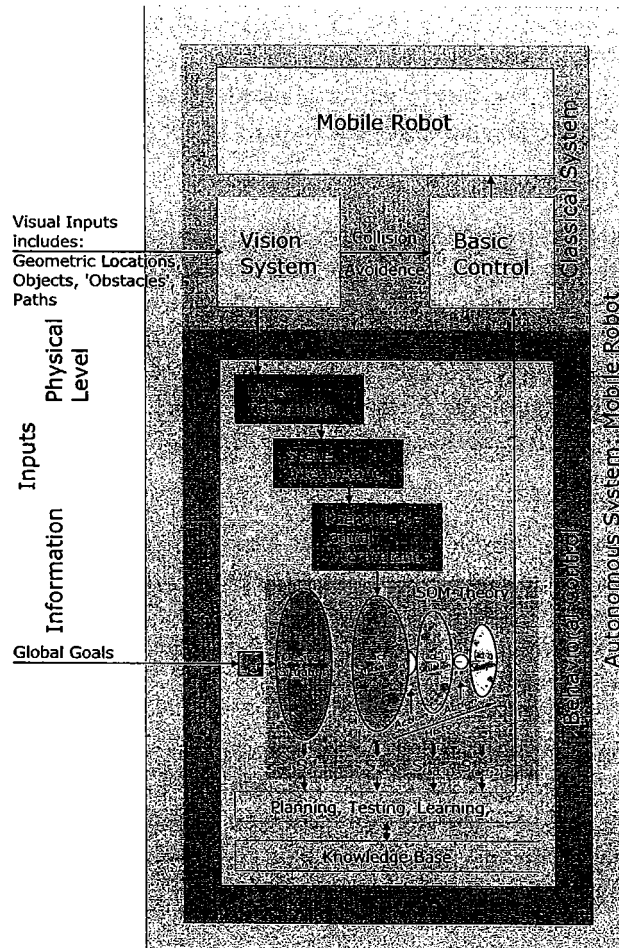


Figure 5. Outline of the proposed intelligent control scheme based on the SOM-technique.

data sets of the knowledge base about known objects, situations, and strategies ('situation' environment) can be used. Therefore the physical oriented input information are clustered, referenced and summarized to a set of characteristics.

Logical Interpretation. The logical interpretation module connects the set of characteristics to known sets of characteristics as known situations identifying the logical structure of the actual situation as new or as known. The result is related to given external goals or even internal part goals etc.

Planning. The planning module has two tasks: to develop strategies reaching external given goals and to observe actual situations and relate them to the previously defined strategy.

For developing strategies the knowledge database has to be used ordering operators, defining sequences, to prepare the acting of AS. The developed strategy

has to be given to the execution level of AS. Observation of the actual situation is important, because of unexpected changes of the environments and of planning errors etc. The detection of difference between the planned reference and the actual situation gives the flexibility to change strategies, to react to unexpected situations and to learn.

Learning. In contrast to classical control schemes and to algorithms an autonomous intelligent system has not only the feature of changing strategies but also of adapting the knowledge background to new situations (and environments). The transformation of spatial (and visual) information into the structural logical information based on the SOM-technique may allow the strong reduction of new information to the relevant source. The algorithms of the learning module (as declared with different qualities) give data for the knowledge base, arranges and changes the knowledge base.

Testing. Besides the case that situations clearly can be identified, not ambiguous situations or relations can be checked using testing strategies. Furthermore it might be necessary to test the environment to learn something about it. This might be a worst case solution, if other solutions are not available. The other necessary aspect can be compared with the learning by playing aspect.

Knowledge Base. Using a SOM-discretized description of the sensor inputs, the developed strategies etc., allowing definition and redefinition of operators etc. the requirements for the technical realization of the data base system are high, this is mainly due to the fact that data sets are dynamically and open, furthermore the assignments are also dynamical.

7. Summary

The contribution briefly describes elements and architecture of a new kind of intelligent control to autonomous systems like mobile robots based on the description of the Human–Machine-Interaction.

References

- Cacciabue, P. C.: 1998, *Modelling and Simulation of Human Behavior in System Control*, Advances in Industrial Control, Springer, New York.
- Gigerenzer, G. and Goldstein, D. G.: 1996, Reasoning the fast and frugal way: Models of bounded rationality, *Psychological Review* **103**(4), 650–669.
- Görz, G. (Hrsg.): 1995, *Einführung in die künstliche Intelligenz*, Addison-Wesley.
- McCarthy, J. and Hayes, P. J.: 1969, Some philosophical problems from the standpoint of artificial intelligence, *Machine Intelligence* **4**.
- Müller, J. (Hrsg.): 1993, *Verteilte Künstliche Intelligenz*, BI-Wissenschaftsverlag.
- Ogata, K.: 1990, *Modern Control Engineering*, 2nd edn, Prentice-Hall, New Jersey.
- Sandewall, E.: 1994, *Features and Fluents*, Clarendon Press, Oxford.

- Schweitzer, G.: 1970, Probleme und Methoden zur Untersuchung des Regelverhaltens des Menschen, in: W. Oppelt and G. Vossius (eds), *Der Mensch als Regler*, VEB Verlag Technik, Berlin, pp. 159–238.
- Söffker, D.: 1999, Observer-based measurement of contact forces of the nonlinear rail-wheel contact as a base for advanced traction control, in: J. Wallaschek, J. Lückel and W. Littmann (eds), *Mechatronics and Advanced Motion Control, Proc. of 3rd International Heinz Nixdorf Symposium*, HNI-Verlagsschriftenreihe, Band 49, pp. 305–320.
- Söffker, D.: 1999, Modeling the human-machine interaction, Technical Report SS-99-04, Knowledge System Lab, Stanford University, California.
- Söffker, D.: 2001, System-theoretic modeling of the human interaction with technical systems, Habilitation Thesis (in German), University of Wuppertal.

