Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®
Robotics and Autonomous Systems 46 (2004) 151–157
Robotics and Autonomous Systems
www.elsevier.com/locate/robot

Towards genuine machine autonomy

J.-J. van der Vyver [a], M. Christen [a], N. Stoop [b], T. Ott [a], W.-H. Steeb [c,d,e], R. Stoop [a,e,*]

[a] *Institute for Neuroinformatics, University/ETH Zürich, Winterthurerstr. 190, 8057 Zürich, Switzerland*
[b] *Department of Physics, ETH, Zürich, Switzerland*
[c] *Department of Applied Mathematics, RAU, Johannesburg, South Africa*
[d] *International School for Scientific Computing, Johannesburg, South Africa*
[e] *University of Applied Sciences of Northwestern Switzerland, Solothurn, Switzerland*

Received 20 January 2003; received in revised form 8 December 2003; accepted 12 January 2004

Abstract

We investigate the consequences and perspectives resulting from a strict concept of machine autonomy. While these kinds of systems provide computationally and economically cheaper solutions than classically designed systems, their behavior is not easy to judge and predict. Analogously to human communication, a way is needed to communicate the state of the machine to an observer. In order to achieve this, we reduce the proliferation of microscopic states to a manageable set of macroscopic states, using a clustering method. The autonomous machine communicates these macroscopic states by means of a visual interface. Using this interface, the observer is capable of learning to associate machine actions and states, allowing it to make judgments on, and predictions of, behavior. This emerged to be the crucial ingredient needed for the interaction between humans and autonomous machines.

PACS: 05.45.Gg; 05.10.C

Keywords: Autonomous system; Communication; Clustering; Fractals

1. Introduction

In the field of Artificial Intelligence, machine autonomy was originally considered to consist of five aspects [1]:

(1) The ability to make independent decisions based upon observations, to do planning, to draw conclusions and to make judgments concerning consequences.
(2) The warranty of autonomy through guidelines and policies.
(3) The independent completion of tasks, by combining the planning and controlling steps.
(4) The ability to learn and eliminate mistakes.
(5) The ability to cooperate, in particular, with other machines

Due to the more recent methods used for the creation of autonomous systems, notably genetic algorithms, we feel that the concept of autonomy should be formulated as general as possible. Hence, here we define autonomous machines as systems developing

* Corresponding author. Tel.: +41-1 635 3063; fax: +41-1 635 3053.
E-mail address: ruedi@ini.phys.ethz.ch (R. Stoop).

0921-8890/$ – see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2004.01.002

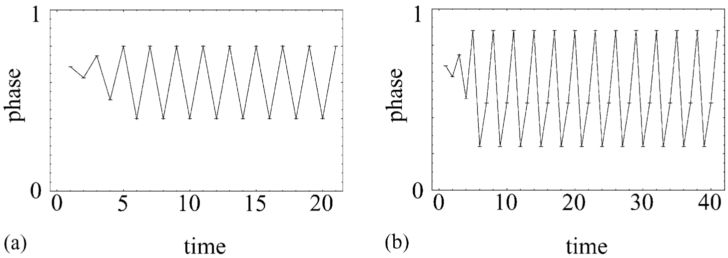


Fig. 1. Autonomous gait generator. By reducing the load, the gait changes from a period 2 (a) to a period 3 (b). The system consists of a chaotic tent map, where the load is implemented by a horizontal line replacing the graph above a certain height (the limiter) [6].

according to their own dynamics, under the interaction with their environment [2,3].

As an example, consider snake-like locomotors [4,5]. Their main class of movements is crawling. However, it is observed that crawling emerges in different patterns, called gaits. For example, some snakes crawl in a sine-like way, while others prefer to move like side-winders. In the context of autonomous systems, this can be interpreted as follows: Take a chaotic autonomous system, for simplicity a 1D chaotic system [6]. Consider the effect of loads, that are omnipresent in nature, on the system. The most obvious effect of a load is that certain areas of the phase space are prohibited and states in these areas are excluded. The effect can be even more dramatic than a mere suppression of some orbits. It has been observed that a previously chaotic system under load may change to stable periodic behavior [7,8]. Moreover, a change of the load often induces a change in the periodicity of the orbit. In the context of locomotion, this means that the load (representing the environment) can be interpreted as a gait selector for the autonomous system, where no explicit control, or adaptation, mechanism is needed to obtain the generically robust gait (see Fig. 1).

This simple example of an autonomous gait generator demonstrates key features of autonomous systems: they may provide computationally and economically cheaper solutions than the classical finite-state machines. Moreover, they have the ability of finding novel, i.e., by humans unanticipated, solutions.

Autonomous systems do away with the explicit controller found in classical rule-based systems (see Fig. 2). The absence of this controller, however, does come at a cost. To enable interaction, the controller is usually used as an interface, reporting the state of the

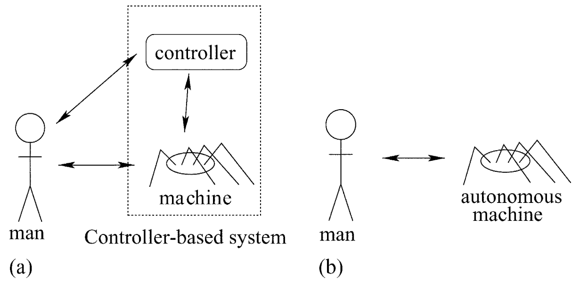


Fig. 2. Comparing interaction with controller-based versus interaction with autonomous systems. (a) Controller-based systems: rule-based, less flexible, resource-demanding; easier to access, due to the existence of the controller. (b) The autonomous system has resource-saving optimal behavior, but is more difficult to communicate with.

system. The lack of a controller, thus, poses a new problem: How can the states and actions of the autonomous system be evaluated, and the behavior predicted? The challenge is to find a minimal encoding/representation of the inner states of the autonomous system, to allow the unbiased judgment of the current state and a prediction of the future behavior, but otherwise minimally binding system resources.

2. Macroscopic states, behavior and communication

The abdication of control of the system in exchange for autonomy has the effect that the relation between internal machine variables and macroscopic machine behavior is no longer inherently obvious. As a consequence, for the representation problem, directly employing the values of the inner variables will be of little help. Example, merely plotting all the (neuron state) values of a neural network does not usefully represent the current state of the autonomous machine. It may be argued that, in principle, any dynamical system's development on multiple time scales could be captured by measuring a scalar time series and using the embedding theorem [9,10]. Still, this does not make the underlying data any clearer. A reduction of the multitude of microscopic states is needed.

Macroscopic states emerge even in the absence of an explicit controller because the internal states are correlated. This is due to the fact that the world perceived by the autonomous systems by is highly structured. Which, in turn leads to clustered microscopic states. The identification of the macroscopic states, however, is an a priori nontrivial task. Recently, however, unbiased clustering approaches have been developed.

That such a clustering approach is in fact feasible is demonstrated in the following example. Consider a system, where its state, at any one time, is a point described in 166 dimensional vector space. A list of 153 such vectors is analyzed using the superparamagnetic clustering algorithm [11]. In this approach, the tendency to cluster is counteracted using an order parameter, i.e., the order parameter is used to break up the data into smaller clusters. The successful reduction from 166 microscopic variables to eight macroscopic states is shown in Fig. 3.

We have shown that macroscopic states emerge from the interaction between the system and its environment. This leaves the question of how these states can be communicated to an observer. Symbols is the most obvious way to express these macroscopic states to an observer. Most often, this is communicated in the form of speech. Verbal man–machine interfaces have been, and still are, the subject of intensive research [12,13]. However, they are generally resource-intensive and not universally understandable. To compensate, in real life, spoken language is often complemented by 'body language'. Visual perception and interpretation of, e.g., human gesture and bearing, to judge another human being's 'state', and to predict future behavior, is everyday practice and considered quite reliable. This is formalized in the facial action coding system [14] used in psychology. The relative changes in a small number of facial

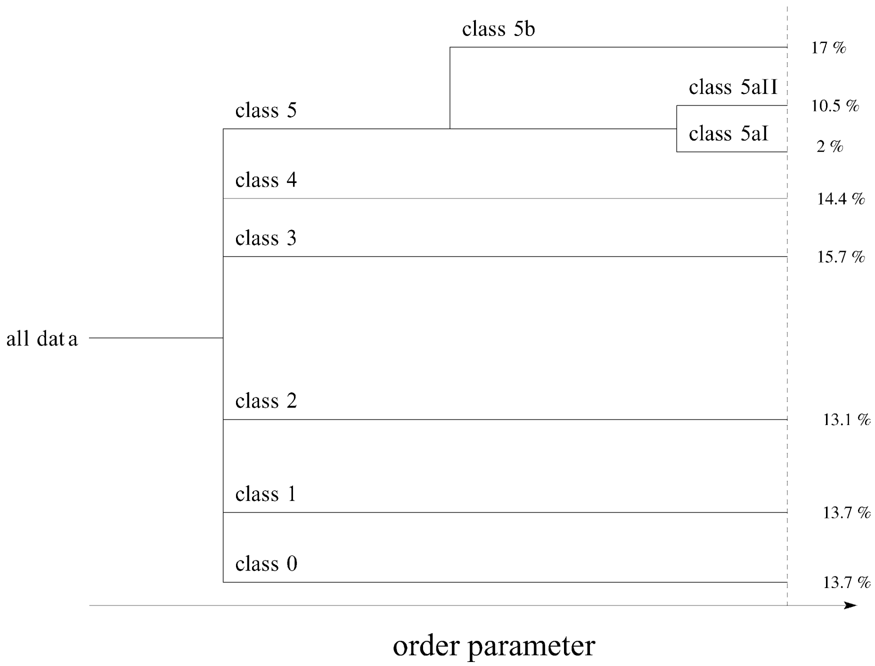


Fig. 3. A clustering dendrogram. As the order parameter increases along the *x*-axis the data separates into clusters. The percentages indicate the number of points per cluster.

landmarks reliably communicate the inner psychic state of the patient [15]. Visual cues have proven a practical and efficient way to communicate macroscopic states.

3. Visual communication

Following this reasoning a visual interface is the most reasonable, but it does not delimitate how this can be realized. First and foremost, these state representation should not be anthropomorphic, since the biased projection of human-like thinking/emotions onto the machine would be erroneous. An interface that allows humans to (unimpededly) learn to associate the behavior with additional visual stimuli would be ideal. That is, requiring a means capable of representing the macroscopic states through time in a human, and preferably human cultural, unbiased way, in order to facilitate learning.

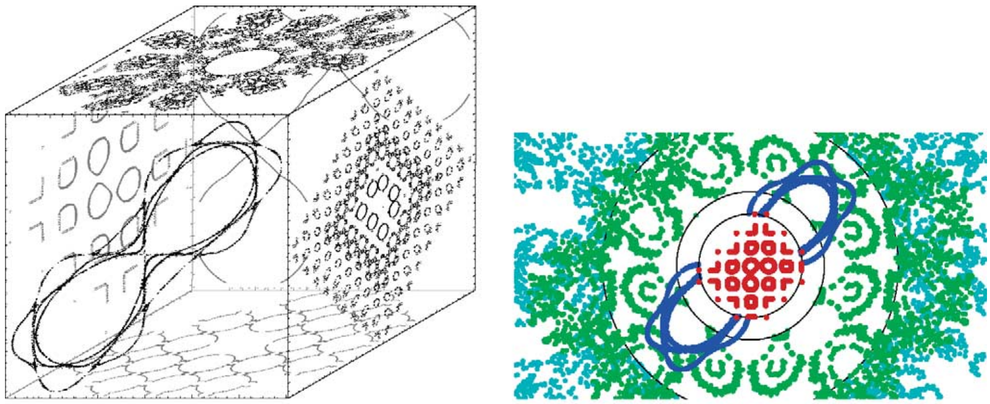
Our solution to these constraints is the use of a fractal pattern generator. Fractal patterns come without predefined meaning and use simple, resource-friendly, generators. As a function of their parameters, they are able to generate a large variety of patterns. Due to their self-similarity properties, they allow a fast grasp of essential pattern structures, and have the potential to adjust to changing macroscopic state compositions. While the patterns for given parameter values are without precise predictability, they exhibit an overall continuity property. This property is helpful when behavior should be associated with the succession of states.

For example, a very simple fractal pattern generator is given by the rule $k_{t+1} = l_t - \sqrt{|bk_t - a|}\,\mathrm{sign}(k_t)$, $l_{t+1} = k_{t-a}$, with parameters $\{a, b\}$. The fractal map, despite of its simplicity, generates a huge variety of geometries and temporal paradigms. By associating each state with a parameter pair $\{a, b\}$, the generated patterns can be used to represent these states. Depending on the number of independent parameters, more complicated pattern generators can be used. Care, however, must be taken to ensure that the dependence on all the parameters is of comparable impact, in order to obtain a transparent representation of the main states and the system behavior.

Such a visual interface was implemented as a contribution to the 'Ada – the intelligent space' [16] exhibit of the Swiss national exhibition 'EXPO.02'. This exhibit had as the main goal the initiation of a public debate on the application, and implication, of brain-based technologies [16]. As such, its original design was that of an autonomous system, based on a biomorphical neural network connected to sensors and actuators. A simple-to-grasp fractal representation was designed to provide an improved understanding and judgment, by the visitors, of Ada's actions and reactions. Moreover, by the emergent interaction, Ada should develop an autonomous identity on its own, where the interaction with the visitors would replace the role of the friction in the introductory example of autonomy.

The input to his interface was a number of predefined and prelabeled coarse-grained states. The states were grouped on three mutually exclusive axes (labeled 'satisfaction'/'frustration', 'joy'/'sadness', and 'surprise'/'dullness'). Each axis contained three states (labeled 'minus', 'neutral', 'plus'), yielding 27 states in total. For the representation of the states, different 'aesthetic' classes of patterns emergent from the fractal pattern generator were chosen. Forms best expressing the particular macroscopic qualities were selected and put on the axis, where the arrangement from 'minus' to 'plus' indicates the increase of the particular quality, by using the complexity measure of Stoop and Stoop [17] for the available patterns. In this way, a representation as shown in Fig. 4(a) is obtained, where, however, only the extremal states are displayed.

The representation over different time scales needed for an interpretation in terms of behavior was achieved by plotting the fast dynamics within the pattern generation and, on a slower time scale, the change in the macroscopic state. On the fast time scale, in the middle of the screen a freshly generated fractal pattern represents the current state, i.e., present time. On concentrically expanded rings surrounding the inner circle, previously generated patterns are maintained, and handed on from one ring to the next one lying outward, as in the center the current state is rendered. In this way, the succession of states is represented, with the past fading out towards the frame of the screen. The logarithmically growing size of the patterns supports the illusion of a history passing by the observer. A crude picture of the representation is shown in Fig. 4(b). Whereas the generation of individual points of the central pattern cannot be followed, the overall temporal generation ('temporal paradigm') is perceivable. The update of a pattern containing a few



(a) (b)

Fig. 4. (a) Patterns corresponding to extremal states. (b) Representation layout, showing the succession of four particularly distinct states. Each state is represented by a new fractal being drawn in the center (the red fractal representing the current state). Outer rings carry earlier fractals, partially overlapping with temporarily adjacent patterns. Only the inner deliminators of the rings are shown. The dynamical behavior, which is essential for human perception, is not represented in these figures.

thousand points was typically done in a couple of seconds.

Although Ada was only partially autonomous because of the restrictive time slot allotted for visitors and the generic risk of explorative work during the exhibition, it still clearly demonstrated the feasibility of our approach: even to minimally experienced observers it was immediately clear when the displayed patterns were not determined by Ada's inner states. The displayed patterns simply did not seem to correspond to the behavior of the autonomous system.

4. Conclusions

We investigated the consequences and perspectives of genuine machine autonomy. As such systems lack a controller, an interface for communication has to be explicitly established. Our proposed solution is based upon macroscopic states obtained by means of clustering. These states are visually communicated to the human observer using a fractal pattern generator, providing an unbiased basis for the evaluation of autonomous machine state and behavior. On this basis, communication between the observer and the autonomous system can be established and developed, allowing the system to adapt to a human sociocultural environment.

Our work provides a basis for future work that will allow important in-sights into the structure of optimal human–autonomous machine interaction to be gained. Clustering techniques, e.g., [18], promise a particularly powerful means to further classify internal states for a human observer. As an continuation of our work, an autonomous model system (understood in terms of our definition) will be designed and run in an environment free from prohibitive conditions, evolving along the insights obtained in this contribution.

Appendix A. Technical design aspects of Ada implementation

Inner state representation requires action in real-time, posing heavy demands upon hardware. To some extent, this can be compensated for by careful software implementation, with the consequence that performance requirements largely determine the software implementation design. In our case, this led to the decision to use C as the programming language and SDL (simple direct layer) as the rendering library. For simple interaction between different components of our application, we chose a fast CORBA implementation (ORBit) as the mechanism of communication.

As a consequence, our implementation architecture is client–server based. The software basically consists of (a) a small and fast CORBA server and (b) a number of independent client processes. The software's design allows running these parts on different machines, connected over a standard TCP/IP network. The CORBA server's only task is to hand over the parameter values it receives from Ada to the representation processes. Instead of directly connecting the representation processes to Ada, this architecture was chosen based upon stability and performance considerations (there could, in the future, be more than one such process). Every parameter representation process is a CORBA client which retrieves the current parameters from the CORBA server and calculates the object to be represented.

In the standard Ada implementation, the pattern generator (responsible for the scaling of the current pattern and the calculation of the next one) was the only representation process. It is composed of different threads. One thread simultaneously calculates the patterns from the data received via CORBA and stores them into a pixel buffer as well. The main thread is the rendering thread. It takes the pixel buffer from the pattern generator, scales it according to a time variable and adds it together with the previous pixel buffers to form a 'background buffer'. As a last step, it swaps the current screen buffer with the background buffer.

This implementation assures high frame rates, as the actual rendering process only has to scale, add and blit buffers, avoiding unnecessary communication between threads. Using this implementation on a Pentium IV personal computer, the representation over different time scales poses no problems and is only limited by the human eye's ability to identify pattern structures.

References

[1] G. Strube, Wörterbuch der Kognitionswissenschaften, Klett-Cotta, 1996.
[2] R. Brooks, Intelligence without representation, Artificial Intelligence 47 (1991) 139–160.
[3] R. Brooks, Intelligence without reason, in: Proceedings of the International Joint Conference on Artificial Intelligence, 1991, pp. 569–595.
[4] S. Hirose, Biologically Inspired Robots: Snake-like Locomotors and Manipulators, Oxford University Press, New York, Oxford, 1993.
[5] F. Chernous'ko, The motion of a multilink system along a horizontal plane, Journal of Applied Mathematics and Mechanics 64 (1) (2000) 5–15.
[6] C. Wagner, R. Stoop, Renormalization approach to optimal limiter control in 1D chaotic systems, Journal of Statistical Physics 106 (1–2) (2002) 97–107.
[7] N. Corron, S. Pethel, B. Hopper, Controlling chaos with simple limiters, Physical Review E 84 (2000) 3835.
[8] R. Stoop, C. Wagner, Scaling properties of simple limiter control, Physical Review Letters 90 (15) (2003) 154101.1–154101.4.
[9] H. Whitney, Differentiable manifolds, Annals of Mathematics 37 (1936) 645.
[10] F. Takens, Detecting strange attractors in turbulence, in: D. Rand, L. Young (Eds.), Dynamical Systems and Turbulence, Lecture Notes in Mathematics 898, Springer, Berlin, 1981, pp. 366–381.
[11] M. Blatt, S. Wiseman, E. Domany, Superparamagnetic clustering of data, Physical Review Letters 76 (18) (1996) 3251–3254.
[12] Microsoft Research: Speech Technology. http://research.microsoft.com/srg/.
[13] IBM Research: Human Language Technologies. http://researchweb.watson.ibm.com/hlt/.
[14] P. Ekman, W. Friesen, Facial Action Coding System, Consulting Psychologist Press, Palo Alto, 1978.
[15] J. Endrass, Sequentielle Analyse des nonverbalen Verhaltens in der dyadischen Interaktion zwischen Gesunden und Depressiven, Verlag für Wissenschaft und Bildung, 2000.
[16] K. Eng, A. Baebler, U. Bernadet, M. Blanchard, A. Briska, M. Costa, T. Delbruck, R. Douglas, K. Hepp, D. Klein, J. Manzolli, M. Mintz, T. Netter, F. Roth, K. Wassermann, A. Whatley, A. Wittmann, P. Verschure, Ada—buildings as organisms, in: Game, Set and Match, Faculty of Architecture, TU Delft, 2001.
[17] R. Stoop, N. Stoop, L. Bunimovich, Complexity of dynamics as variability of predictability, Journal of Statistical Physics 114 (3) (2004) 1127–1137.
[18] G. Getz, E. Levine, E. Domany, Coupled two-way clustering analysis of gene microarray data, Proceedings of the National Academy of Sciences of the United States of America 14 (2000) 12079–12084.

J.-J. van der Vyver (IEEE student member) got his B.Ing.EE degree from the Rand Afrikaans University (RAU) in the year 1998, followed by a M.Ing.EE at RAU in the year 2000. During the whole period (from 1994 to 1999) he was a full SASOL burser. He has been registered as a Ph.D. student of the Institute for Neuroinformatics at the Swiss Federal Institute of Technology (ETH), and University Zurich, since the year 2000. His research interests include nonlinear circuits, autonomous systems (embodied cognitive systems), and biomimetic/biomorphic systems (and how it all relates to signal processing, control theory and automation).

M. Christen, born 1969 in Biel, Switzerland, studied philosophy of science, physics, mathematics and biology at the University of Berne. He is currently working on his Ph.D. at the Institute of Neuroinformatics ETHZ/UNIZH. His research focus lies on neural time series analysis and neural coding. He also works as a freelance journalist of science and is interested in the impact of brain-based technology to society.

N. Stoop is studying Physics at ETH Zurich since 2001 and is a member of the 'Schweizerische Studienstiftung'. His academic interests include neuroinformatics, robotics and computational physics. In his spare time, he's implementing neural networks for a Khepera robot or develops web applications.

T. Ott studied physics at the ETH Zurich from 1998 to 2003. Since 2003 he worked as a Ph.D. student in the group of R. Stoop at the Institute for Neuroinformatics (ETH/University Zurich). His research interests concentrate on the combination of statistical physics and neuroinformatics, bio-inspired clustering methods for applications such as sensory fusion as well as questions of theoretical (neuro-)informatics and logics.

W.-H. Steeb got the Ph.D. degree from the University of Kiel and a Dr. habil. degree from the University of Paderborn. He is currently the director of the International School for Scientific Computing at the Rand Afrikaans University. His research interest lie in nonlinear dynamics and quantum computing.

R. Stoop was born 1952 in Davos, Switzerland. He studied Mathematics and did a Ph.D. in computer-aided physics at Zürich University. At the University of Berne, he did a habilitation in the field of nonlinear dynamics. As an associate professor, he lectures at the University of Berne, at the Swiss Federal Institute of Technology Zurich (ETHZ), at the University of Zürich (UNIZH) and at the University of Applied Sciences of Northwestern Switzerland. At the Institute of Neuroinformatics ETHZ/UNIZH, he leads a research group concentrating on the application of nonlinear dynamics and statistical mechanics to problems in neuroscience. He co-authored the book 'Encounter with Chaos' (Springer, 1992). He regularly organizes conferences and is a member of the editorial board of 'Nonlinear Analysis, Real World Applications'