Adaptive and Autonomous Systems and their Impact on Us

Nikola Šerbedžija

Fraunhofer FOKUS, Berlin

Abstract. With technology becoming increasingly ubiquitous and through a wide use of interconnected “smart devices”, the impacts these advanced products have on us is gaining in significance. As technology providers, we are very proud to share the tribute for creating new infrastructures that bring benefits to individuals and society and make life easier. But we may also be held responsible for the possible detrimental impacts that new technology brought about. Especially, if we ignore the threatening consequences and fail to offer some protective solutions. Up to now, there has been some attention paid to privacy issues and security of commercial transactions, but the negative influence of “smart” technology on human behavior has been widely neglected. This paper considers the effects that adaptive and autonomous technologies have on their users. As the impacts can best be observed in practice, a number of application scenarios are taken into account, illustrating both the technical aspects and their possible effects on us.

1 Introduction

As a field initially derived from mathematics and electrical engineering, computer science in its early days performed mostly numeric and data processing in various application domains. Gradually, a number of other disciplines merged into the field, extending digital paradigm with principles taken from physics, chemistry, biology, psychology and social sciences. The application domain also spread to almost all aspects of our activities, making computers unavoidable in industry, administration, commerce and entertainment. Furthermore, present cyber-physical systems are bridging the gap between virtual and real worlds, placing humans directly in the processing loop, increasing the impact the technology has on us. We are not anymore just users that exploit the benefits of the technology, we ourselves are becoming a part of the computation and thus exposed to possible manipulation. More and more often, we rely on computers and do what they instruct us to do. We are guided by the rules imposed by technical systems, modifying our behavior accordingly. Mostly it is done with the user consent, but not rarely the changes in our behavior happen unconsciously.

This paper focuses on adaptive and autonomous systems as an important segment of technology advances and investigates the effects that practical deployments may have on us. Engineering the natural phenomena like awareness, adaptation and autonomy does include a human in the processing loop affecting and controlling personal behavior, causing impacts that may go beyond our ethical norms. Having in mind that artificial adaptation and autonomy are not the same as the natural ones, the challenges, difficulties and solutions for development and deployment of adaptive and autonomous systems are explored. The reasoning described here stems from two concrete methodologies and their applications. The two approaches are presented rather descriptive, focusing on a general strategy necessary to explore technology impacts.

Further references for the detailed technical solutions could be found at [1, 3]. One case study is specially highlighted: the cyber-race, where our autonomous robot competes with a human and wins. The example helps in better framing the rationale of use, indicating in which domains we should or should not apply autonomous systems and to what extent we are sacrificing our own autonomy when using autonomous systems.

The paper is structured in six sections. After introduction, the section 2 considers technology advances and their possible influence on people and society. Then, the two approaches for adaptive (section 3) and autonomous (section 4) systems are presented together with a number of concrete use cases. A representative deployment scenario of autonomous system is given in more details (section 5) and the concluding remarks (section 6) highlight major conclusions and trace further work.

2 Technology Impacts: A Digital Storm

Brainstorming is a popular phrase among scientists. It denotes an intense exchange of ideas on a specific research challenge. But brainstorming our brain in order to understand it and engineer something similar, is like going through a labyrinth of *controversy*. Neural networks, artificial intelligence, psychological and evolutionary computing are examples of the domains with numerous breakthroughs, but no real "silver bullet". In a way, this is understandable since we are dealing with extremely hard problem. If our brain were simpler, we would be simpler too, and it would be even more difficult for us to comprehend how it functions and engineer something similar. Even when a natural characteristic is successfully re-created, its deployment often brings dual effects. For example, reasoning about autonomous systems and their impacts, brings us in a dilemma: "should we use the autonomous systems, and if yes, in which domains". Namely, if we define autonomy as "acting according to own rules", a question arises: are we autonomous when we use autonomous system? Obviously a use of autonomous systems requires giving up a part of the own autonomy. Considering a wider arena of smart technology, one could observe a resembling situation in today market: there are numerous advertisement of a kind: "be smart use smart systems" luring people to purchase some smart product. Mostly the people who do not doubt their own reasoning fall in the trap. Sometimes, an artificial achievements may weaken its natural counterpart, not intentionally, but rather as a side effect. It seems that brainstorming the natural phenomena resulted in a digital storm, characterized by a wide and premature acceptance of digital technology, where we cannot be only proud of its positive impacts but also held responsible for some of its negative consequences.

Many technical breakthroughs stem from imitating the nature and its spectacular wonders. But how can it be, that imitating something so harmonious can result in controversy? Probably, it is our inappropriate interference with nature that creates puzzling situation. Concentrated on a specific engineering goal, we often overlook the wide spectrum of the problem, solving one aspect and neglecting the others that constitute the natural harmony. It seems that we are impatient: we try to solve the problem before we truly understood it. For example in the artificial intelligence or neural networks domain an engineer interferes with something, not fully understood even by neuro-experts. It seems that we are dealing with multifaceted assignments, appealing but hard, maybe Sisyphus's. The controversy often occurs when advanced computing model includes humans into the processing loop creating obvious benefits and hidden harms. Especially when interfering with subtle subjective experiences like senses, feelings or (sub) consciousness. Essentially, we are all very skeptical in submitting our autonomy, privacy and habits to the others (especially machines). Practically, however, we have been guided, data minded and exposed to commercialization to the point that we do not see the obvious anymore.

Since 1995, the famous Weiser's statement on disappearing computers [4] has been inspiring the work in ubiquitous computing domain. It envisages that "computers will be disappearing i.e. they will be miniaturized and integrated in fabrics of everyday life, omni-present and invisible". In 2005, Streitz and Nixon further elaborated: "The rate at which computers disappear will be matched by the rate at which information technology will increasingly permeate our environment and determine our lives" [6]. The man-computer interaction will intensify to such an extent that even absence of interaction will gain an interaction significance (scaring but true). In another words you cannot not communicate in inevitably ubiquitous environment (the statement known as the first of five /sometimes paradoxical/ axioms of communication, from Watzlawick [6]). It seems the better we are in our technical achievements, the controversial it gets. Yet another paradoxical situation could be observed on the Internet arena. Once conceptualized as a highly distributed network, World Wide Web is being more and more controlled. Some of the leading Web2.0 players are squeezing the Web into an almost centralized system with millions of thin clients communicating with a few major sites, that collect and process data (often silently and without user consent) for commercial purposes.

Being aware of promises and threats that modern technology has brought about we should strive to enhance the benefits, reduce (collateral) damages and avoid controversy. Otherwise we have not learned much from the past. Back in 1818, M. Shelley[7] frightened the generation to come with her gothic Frankenstein tale about curse of man using technology to "play God". Today, modern "Smart-kensteins" are devouring our privacy, social spirits, and physicality, according to Huxleys [8] nightmarish observation on how control can be achieved by giving people what they want. We are stunned, but instead of adapting our high-tech products to us (our norms and practices) we are changing our behavior according to the technical systems rules: we are re-defining our ethics and conducts (e.g. often we hear that privacy is not that important as traditionally thought; virtual contacts are as important as physical; etc,), letting our wash-prone brain to adapt to a new situation. And being so naturally adaptive it will continue to do so.

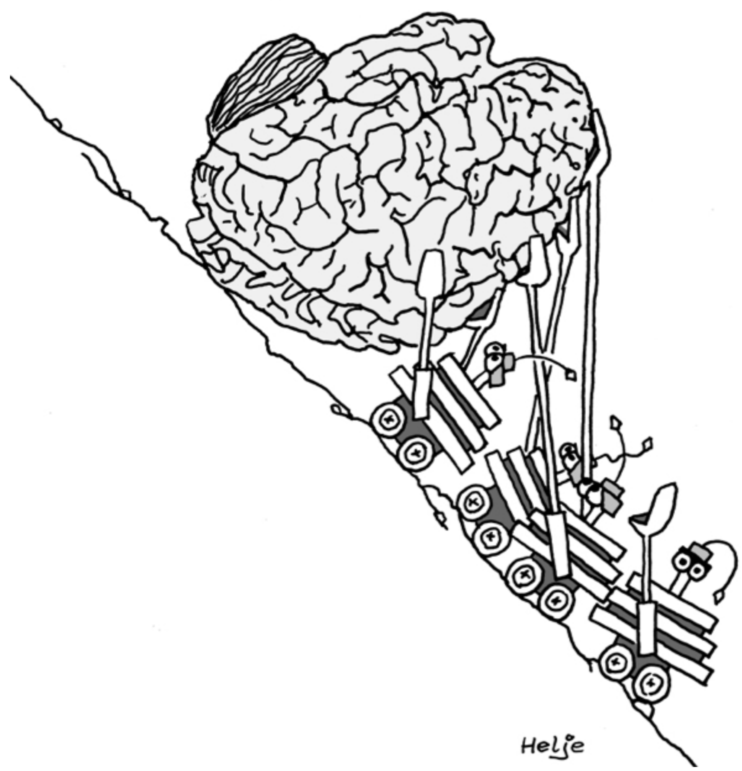

Fig. 1. Smart Sisyphus

Aware that a success in engineering a natural phenomenon may be a Pyrrhic victory for humanity, especially when carelessly mass-applied on individuals and society, we should try to explore both the principles of adaptive and autonomous systems and possible impacts of their use, thus maximizing the benefits and minimizing drawbacks and paradoxical traps. Not to repeat the mistakes from more mature Internet domains where the impacts are being discovered a *posteriori*, here the possible drawbacks are considered in time, hoping that they could be avoided.

3 Adaptive Systems

One approach to construct a user-centric adaptive systems is the reflective technology [1]. It is an interdisciplinary endeavor aiming to construct smart environments with pervasive adaptive control. A general purpose assistance should enrich control systems with implicit man-machine interaction sensitive to cognitive, emotional and/or physical state of the user. The ultimate goal of such reflective assistance is to observe people in a specific real-life situation, diagnose their psychological and behavioral state and influence the ambient accordingly. The system uses reflective technology to exercise pervasive adaptation through non explicit man-machine interaction based on context awareness. Recent results demonstrates its effective use in vehicular domain and promise further applications in environmental, ambient assisted living and health care areas [1].

3.1 Reflective Technology

In effort to mimic the adaptation process, as it appears in the nature, and to apply it within man-machine interaction, reflective approach deploys the biocybernetic loop to make users psychophysiological data a part of computer control logic. The function of the loop is to monitor changes in users state in order to initiate an appropriate computer response. This approach extends the original concept to a wider set of input information (e.g. social and behavioral) allowing for a composite analyses and decision making. It also takes results of affective/physiological computing and combines it with high level understanding of social and goaloriented situations. Bio-cybernetic loop [2] is implemented with the help of sense-analyse-react control troika. Firstly, reflective ontology classifies numerous factors that determine users states, social situation and application goals, defining elements for decision making. The ontology is then expressed in a number of XML-based taxonomies that allow for a uniform deployment in data acquisition, users state diagnoses and activation of corrective actions.

Reflective framework is service- and component-oriented dynamic and reactive middleware that runs multiple bio-cybernetic loops featuring pervasive adaptation at different time scales. The software architecture is layered as follows:

- Tangible layer - a low-level subsystem that controls sensor and actuator devices. It offers its services (sensor measurements/actuator controls) to the rest of the system.
- Reflective layer - a core of the system that combines tangible services with users profile and scenario descriptions to perform diagnoses of users state and provoke system (re-) action relative to the situation and the application goals.
- Application layer - a high level part of the system that defines application scenario and system goals. By combining low and high level services and components from other layers, application layer runs and controls the whole system.

The reflective framework has been developed using the software components paradigm and implemented in the Java programming language on top of OSGI environment.

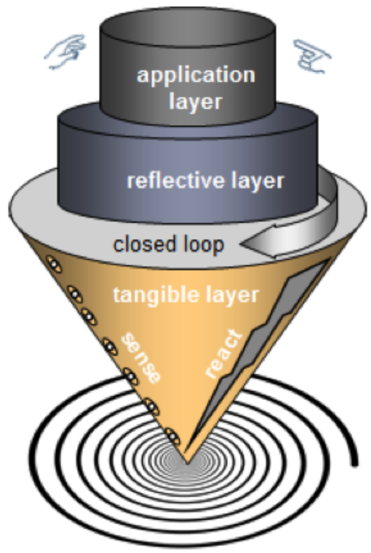

Fig. 2. System architecture

Figure 2 illustrates the reflective software architecture with three major stages, exercising different bio-cybernetic loops at tangible, reflective and application level. The control loop (initialized with users profile and scenario settings denoted by human hand on the top of Figure 2) starts by sampling the psycho-physiological measurements, continues with their analyses and finishes by adaptive system reaction (denoted by a "digital hand" on top of Figure 2). In a next iteration the system influence (caused by the reaction) can also be sensed and further tuned.

3.2 Reflective Use Cases

The developed reflective system is able to grasp and influence human mental and physical state. Together with psychologists [2] and bio-engineers [1], we implemented a personal advertiser that could tailor advertising content to the mental reaction of the viewer; a mood player that selects the music to comfort the listeners emotional state; a computer game that adapts its difficulty level according to mental effort of the player, a smart seat that recognizes how comfortable it is for a person and re-shape accordingly. Putting some of the stated results together, we prototyped a "vehicle as co-driver" [1] system with a capability to configure and dynamically modify vehicles settings and performance according to emotional, cognitive and physical drivers condition. When deployed, a reflective system is personalized and its reactions are tuned to the person being observed. In a number of case studies a positive impact in terms of comforting users physical emotional and cognitive situation have been experimentally proved [1]. However, a spectrum of possible misuses is also very wide. An adaptive advertiser could mislead a consumer, a mood player could manipulate emotional state of a person, adaptive computer game could exhaust a passionate player and a wary "co-driver" silently collects and maintains personal information that the driver would not necessarily share with others.

4 Autonomous Systems

Within the ASCENS approach [3] we developed numerous tools and methods that support one or more development life cycles of autonomous systems, from requirements specification up to run-time monitoring and verification. Here we choose ethically neutral terrain, deploying awareness and autonomic behavior purely in software, making our systems "aware" of their functional and operational requirements, performance and surroundings. However, keeping a human out of the processing loop has not saved us completely from the ethical concerns. The impact there is less direct as in affective user-centric systems, but relying on system autonomy does indirectly affect our own autonomy.

4.1 ASCENS Technology

ASCENS approach explores awareness, adaptation and self-organization offering high-level methods and practical tools for developing autonomous systems. Under the motto simplifying complexity, the technological challenge is in controlling the dynamics of inherently distributed environments while harmonizing and optimizing individual and collective goals. Trail examples are on: how to organize a rescue operation with self-aware and self-healing robot swarms; how to build a scientific cloud platform that turns a huge number of voluntary computing devices into a super computer; and how to optimally control mobility with electric vehicles taking into account energy restrictions, multiple itineraries, parking availability and traffic conditions.

To behave autonomously, a control system needs to maintain knowledge about itself (particular objectives, capabilities, execution state and restrictions) and about its environment. Such collection of facts yields awareness of own functionality and effects it has on the environment which further allows for adaptive performance. Being capable of operating according to these three principles (knowledge, awareness, adaptation), the system is able to re-configure, re-tune and act appropriately thus behaving in autonomous manner.

The approach breaks up a complex control problem into its elementary constituents. It deals with complications at a bottom level, solving issues at a lower scale and then harmonizing these solutions with more global ones. Localization and de-centralization is the fourth major principle of our approach. Service components with clearly defined elementary objectives are basic system elements. They gather in larger symbiosis called ensembles in order to fulfill collective goals. As the controlled situation changes, i.e. goals are (partially) fulfilled, re-grouping takes place and the symbiosis re-structures. The criteria to construct an ensemble of service-components is some joint interest which can be expressed as a logical sentence, e.g. "connect all robots that can carry up to 4kg and are in the radius of 100m with the aim to cooperatively transport 25kg heavy object" or "select all free parking lots in the radius of 300m that have a charging plug". That makes the communication implicit and predicate-based. The connections are established at run-time, depending on the live situation at particular time. These logical rules for highly dynamic grouping are further used for formal reasoning on optimization and coordination among distributed elements.

The overall system development life cycle consists of the following phases: rigorous design (requirement specification, modelling and validation/verification), deployment (programming) and run-time monitoring (live examination of awareness, adaptation and autonomous behavior). A number of tools have been made [3] that support the development process at each step, thus guiding and facilitating the whole development process.

Requirement specification is a phase where the dissection of the problem to be solved takes place. Each system element is separately defined both functionally (what to do) and non-functionally (how to do) yielding a set of goals that embrace the terms of functioning and description of environment. The knowledge required for system awareness and adaptation is used as a major attribute repository for system construction. The SCEL (service-component ensembles language) [9] has been developed for high-level system modelling with service components and their ensembles. Both service-components and ensembles have local knowledge used to express their goals. Knowledge is represented by ontologies that contain hierarchical and meaningful description of system properties and system goals. The goals are described as rules i.e. logical expressions with system properties.

The adaptation phenomenon is formally modeled as a progress in a multi-dimensional space where each axis represents one orthogonal aspect of system awareness (facts about its own functional, operational, or any other necessities defined within requirement specification phase). Adaptation actually happens when the system state moves from one to another position within the space according to the pre and post- condition on each of its awareness- dimensions. Adaptation is a continuous process where a system acts appropriately i.e. in harmony with own capabilities and the observed environment. The adaptation model called SOTA (State of the Affairs) [10] is used to extract major application requirements and offer appropriate adaptation patterns that effectively control system dynamics with numerous feedback-loops.

In order to guarantee correct and timely behavior in such demanding and highly dynamic circumstances this approach relies on formal methods. The major safety and liveness properties are formally proved using SCEL process algebra (e.g. prove that two e-vehicles will never block each other while competing for a free charging station, or prove that the foraging algorithm of a robot converges in a given time). Further validation and verification of specific optimization algorithms are performed in order to guarantee correct system behavior in early design phase (e.g. prove that the optimization method will deliver the most energy-efficient route for a given multi-routing problem). Once the system is rigorously modelled and validated, the actual deployment may take place sewing the system together. The jRESP[3] and jDEECO [3] deployment tools offer direct Java programming support for the SCEL and SOTA models. Due to a seamless functioning of autonomous systems, where system changes are means for "appropriate" behavior, possible malfunctions are difficult to discover. Therefore, a number of tools have been developed for run-time monitoring where internal system knowledge and topology (ensemble construction) as well as awareness and adaptive characteristics are observed. For example, the monitoring tools can visualize how the robots, close to the target and with enough battery-charge are grouped into ensemble to perform joint transport of a heavy object. Once the task is performed, the ensembles are dismantled freeing robots for another assignment. Monitoring inspects and displays major system principles: knowledge, awareness and adaptation, offering a visualization of dynamic ensemble building criteria, thus directly observing autonomous behavior. If some malfunctioning is discovered at run-time, a system modification is considered going back to modelling and design system development phases.

4.2 ASCENS Use Cases

Pragmatic orientation means building technical systems that perform concrete tasks, like autonomous robot swarms performing rescue operation, autonomous cloud platforms transforming numerous small computers into a super computing environment or autonomous e-mobility support that ensures energy-aware transportation services. These are highly dynamic environments where physical and social context, operational and functional requirements and workloads are constantly changing. Through practical deployments we want to show that our approach behaves autonomously and at the same time it integrates smoothly into our ethical codex. Since autonomous behavior means functioning without human intervention, we need to ensure that rules driving system autonomy does not collide with our own independence.

By design our robots know their own functionality, battery state, position, and the location of nearby robots, so they indeed perform actions according to that knowledge. The collective knowledge is constructed dynamically based on the contextual situation, making the whole system highly cooperative, self-healing and autonomous. In the cloud computing scenario, the system knowledge is based on the computing capabilities of voluntary computers that join the cloud. Most of the control functions are exercised at ensemble level featuring self-monitoring, self-organized and distributed deployment and execution. The control system for e-mobility takes into account on-going information about routing, vehicle energy states, traffic conditions and parking/charging availabilities and calculates optimal routing. The system is highly dynamic with ensembles, representing different system goals, being continuously (re-)organized offering alternatives for optimized control (like ensembles of near-by parking places, ensembles of vehicles travelling in same directions, etc.). In a live system, adaptive behavior is crucial as sudden changes in real settings require re-optimization and re-allocation of resources.

The autonomy model we are using is based on presupposed knowledge and real-time awareness and adaptation. Contrary to natural autonomy which is based on learning, experience and evolution, here, the autonomy is much more rigid, based on prior system knowledge and corresponding rules and policies. It is restricted to our understanding of the control environment, without support of (artificial) intelligence, learning, or similar strategies that would ensure skillful handling of unknown problems. Such systems should be used only where human control is not possible due to different dangers, heavy tasks, compute intense calculations, etc. or when delegating a task to a machine is a genuine benefit for people. The application area should comprise well known domains where problems to be solved can be predicted or their solution depends on huge number of factors that is overwhelming for us (e.g. traffic conditions, multi-route optimization, scientific computing, etc.). Further application domains include operations in places dangerous for humans (emergency situations), or assisting by disabilities (medical assistance systems), or even doing boring tasks for us.

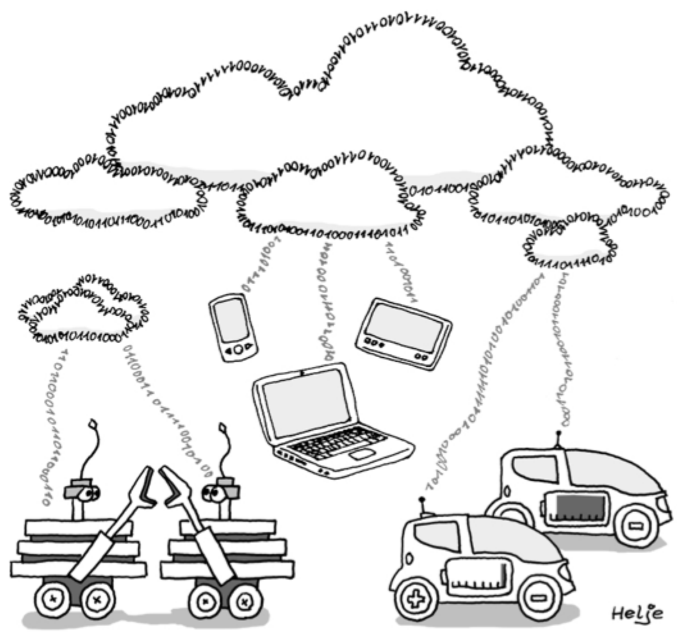

Fig. 3. Digital clouds

The cases where autonomous systems may interfere with our own autonomy should be considered carefully. In general, where a person is a part of the control loop, possibilities of negative impacts are higher: the autonomous system may pursuit the goals which are not necessarily the same as those the person using autonomous system wants to follow.

5 Cyber Race

The challenges of controlling the robot behavior in performing certain task can best be understood if seen from the robot perspective. The complexity does not primarily come from the task itself, but rather from the interaction that goes on between the robot sensory system, environment and self-directed robot performance. To illustrate that, we set an exhibition at well attended ICT conference [11] where our autonomous robot competes with a human-controlled robot. The task is to find building blocks in a closed area, grab them (one by one), and carry them to the place where a wall should be constructed. The robot is operated by a joystick which can move the robot left/right; forward/backwards and instruct it to grab/release the building blocks. The task seems trivial, so most of the competitors believe our robot does not stand a chance. That proved to be wrong. Most people lost, only a couple of young, joystick-virtuous competitors won. Then, we imposed a "fair-play" rule: since the robots sensory system is less sophisticated than ours, we reduced the vision of the human competitor to the visual system of the robot, giving the competitors equal chances. When both competitors have exactly the same information about environment, our robot performed much better. That shows how seemingly simple assignment (from our point of view) is actually complex for a fully autonomous robot. Taking into account relatively primitive robot sensory system, the robot performance is quite good and reliable, especially well-suited for the kind of tasks that we do not want to participate in (like removing objects in a poisoned/radiated/high temperature areas, carrying heavy objects, underwater operations, etc.) [12].

6 Conclusion: Sunshine Breaks through Digital Fog

The illustrations presented summarize the approach in dissecting and sewing adaptive and autonomous behavior. The first figure shows a swarm of autonomous robots rolling the human brain up the hill, a task too difficult to complete. It illustrates the way out of one of the hardest and most controversial assignments men can get. Instead of doing the task ourselves, we can assign it to a multi-robot system. Metaphorically, the figure also expresses our belief that overthrowing human brain is a Sisyphus task. The second figure illustrates reflective approach to create a personal assistant that adapts to physical and mental state of a user. Iterative and ubiquitous nature of the system is metaphorically presented by a spinning top that sense, diagnose and react in each of its rounds (being driven by measurement and computer generated actions). A dazzling

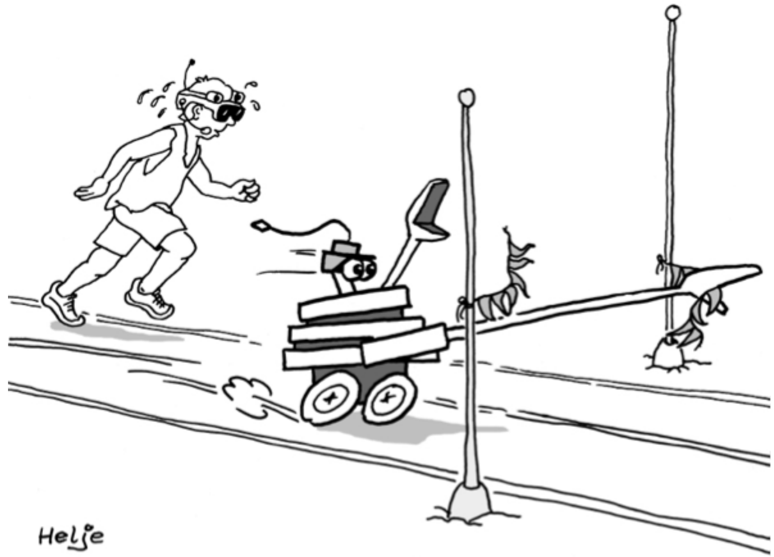

Fig. 4. Cyber runners

character of the approach that may influence person involved, both positively and negatively, is illustrated by "magic circles" casted by the spinning top.

The third figure illustrates our approach to decompose a complex system into simple elements, structure them in digital symbioses (depicted as clouds) and use them to control swarm of robots, electrical vehicles transportation or to manage the cloud computation itself. These are all ethically non problematic practices, delegating a part of tasks we are either not willing to get involved in, or not capable to solve, to a technical system. In this case: (1) rescue operation in presence of dangerous material, (2) optimization of multi-route problems in presence of energy restrictions and (3) performing scientific calculations on a cloud platform. The whole approach features de-centralization, symbiotic grouping and implicit communication. The resulting system behavior is autonomous based on local/global knowledge, awareness and adaptation. The absence of central control is crucial for our approach and in all cases the application scenarios run in a truly distributed manner.

The fourth figure metaphorically shows our competition arena [11] where an autonomous robot wins the race against its human rival. The cyber runner experiment places the human competitor into "the robot perspective" and demonstrates how a cyber-vision may reduce our skills. Often, cyber glasses are advertised as something that "extend" our perception which is not necessarily always the case. Besides offering extra information, they can also decline our concentration, sometime mislead and in general weaken our performance.

The times of paradoxical "centralized network" may be passing as we are adding more and more smart devices into the network that cannot be effectively managed centrally, but is rather self-organized and performs in an ad hoc manner. Evolving cyber-physical systems are melting man-machine interaction into a man-machine confluence where protecting human virtues and ethical codices could be imbedded into system requirements level and more effectively supported in all development phases. The near future technology may bring new generation of simple and highly de-centralized autonomous devices that self-assemble (dis-assemble) in fulfilling a temporal goal. Businesses may look for new methods of technical symbioses with dynamic grouping and dismantling, which would perform tasks autonomously and much more efficiently. Big data mining and collecting everything about anything may lose its significance, because new generation of systems will work much more effectively if bypassing the overloaded and busy Internet, concentrating rather on qualitative then on quantitative data analyses. Re-gaining our privacy and social norms may be a collateral benefit, coming as a sunshine through the digital clouds.

Acknowledgements. All the credits for the work described here goes to the REFLECT [1] and ASCENS [3] project teams. However, if there is a single person to be highlighted as a pillar of the research being conducted, then it is Martin Wirsing [13]. He and his LMU team inspired and contributed to the two projects significantly. Working with Martin was like balancing at the pinnacle of the computer science breakthroughs, both fascinating and provocative at the same time. The more successful we were the more controversial it got. This paper recalled some joint achievements, thoughts and afterthoughts steaming from our collaboration and is dedicated to Martin Wirsing. The tribute for the cartoons on figures 1, 3 and 4 go to Jens-Helge Dahmen (Fraunhofer FOKUS Berlin), who designed them for the purpose of this paper.

References

[1] REFLECT project - Responsive Flexible Collaborating Ambient, available at, http://reflect.pst.ifi.lmu.de/
[2] Serbedzija, N and Fairclough, S. Reflective Pervasive Systems. ACM Transactions on Autonomous and Adaptive Systems (TAAS), Vol. 7 (1), April 2012
[3] ASCENS Project - Autonomic Service-Component Ensembles, available at http://www.ascens-ist.eu/
[4] Weiser, M. The computer for the 21st century. Scientific American, (Sept.1991), 94104
[5] Streitz and Nixon (2005). The Disappearing Computer, 'Communications of the ACM, Vol.8,No.3 2005.
[6] Watzlawick, P., Beavin-Bavelas, J., Jackson, D. 1967. Some Tentative Axioms of Communication. In Pragmatics of Human Communication - A Study of Interactional Patterns, Pathologies and Paradoxes. W. W. Norton, New York.

[7] Shelley, M. Frankenstein. Ed. Johanna M. Smith. Boston: St. Martins, 1992. (The first edition of Frankenstein; or, The Modern Prometheus was published anonymously in three volumes by Lackington, Hughes, Harding, Mavor and Jones on 1 January 1818.).
[8] Huxley, A. Brave New World, New York, London, Harper and Bros., 1946.
[9] De Nicola, R., Loreti, M., Pugliese, R. and Tiezzi, F. A formal approach to autonomic systems programming: the SCEL Language. ACM Transactions on Autonomous and Adaptive Systems. pp. 1-29. ISSN 1556-4665 (In Press) (2014)
[10] Abeywickrama, D.B., Bicocchi, N. and Zambonelli, F. SOTA: Towards a General Model for Self-Adaptive Systems, WETICE, pp.48-53, 2012 IEEE 21st International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2012
[11] Serbedzija, N. The beauty is in the eye of the beholder, blog at http://blog.ascens-ist.eu/2013/11/beauty-is-in-the-eye-of-the-beholder/
[12] Soleymani T., V. Trianni, M. Bonani, F. Mondada and M. Dorigo. Autonomous Construction with Compliant Building Material. Proc. 13th International Conference on Intelligent Autonomous Systems IAS-13, Advances in Intelligent Systems and Computing, Vol. 301, Springer, 2014.
[13] Martin Wirsing, http://www.pst.ifi.lmu.de/people/staff/wirsing/