EUROBOT:2008
MISSION TO MARS

EUROBOT CONFERENCE 2008

Ten Years of Experience with Eurobot ; Achievements, Lessons Learned and General Comments

Jean-Daniel.Dessimoz@heig-vd.ch

Eurobot 2008 – International Conference on Robotics Research and Education,
Heidelberg, Germany, 22-25 May 2008, ISBN: 978-80-7378-042-5

http://www.heig-vd.ch, http://lara.heig-vd.ch

eurobot

Hes·so Haute Ecole Spécialisée de Suisse occidentale

heig-vd Haute Ecole d'Ingénieurs et de Gestion du Canton de Vaud

cetT

institut d' Automatisation industrielle

Ten Years of Experience with Eurobot ; Achievements, Lessons Learned and General Comments

Jean-Daniel Dessimoz[1]

[1] HESSO-Western Switzerland University of Applied Sciences, ,
HEIG-VD, School of Management and Engineering,
CH-1400 Yverdon-les-Bains, Switzerland
Jean-Daniel.Dessimoz@heig-vd.ch

Abstract. After 10 years of participation to Eurobot, time has come to sketch a summary of knowledge and expertise gained. Eurobot has brought on the world scene, for students in engineering schools and fans of technology, a unique opportunity to design autonomous mobile robots, most often with excellent agility and dexterity properties. Eurobot typically sets two major challenges: design of fully autonomous, mechatronic systems; and design of a powerful programming and control environment for very numerous and fast adaptations. Here mechatronic systems require a lot of original design, because of requirements yearly renewed and because of a restricted volume allowed. As a programming environment, we have created Piaget, which is very simple, effective, and quite constant for the final user, even though the implementation has changed a lot through the years, in terms of implementation language, operating system, and control architecture. Education and research have both benefited from Eurobot initiative. For education, in addition to the positive effect on students directly involved, Eurobot has contributed to a much larger impact on society, notably at schools and in the training of the youth. In research, we have in particular found that improvements in quantitative cognitics, and considerations about the relative agility of agents in control loops were important contributions to be fostered. The paper still brings a number of other remarks emerging from a decade of Eurobot experience.

Keywords: Autonomous mobile robots, Cognitics, Edutainment.

1 Introduction

Our university has joined the Eurobot initiative and competitions from the very beginning, in 1998. For a few years before (1994), the movement had been launched within France boundaries. And as a matter of fact, Switzerland initiating in 1998 their own national competition, and joining later on for the final round at the international level, contributed to turn the previously French-only competition to a truly international event. Eurobot was born.

Many documents exist about Eurobot ; for some years, 15 countries and more had television reports about competitions. Euronews has broadcast some reports, and especially the French television channel M6, with its famous E=M6 program, has regularly broadcast reports about Eurobot competitions. The European Community has also helped disseminating results and bridging the gap between Science and Society at large, by sponsoring Eurobot. An excellent presentation of Eurobot competitions is available in [1].

Fig. 1. An excellent six minute description of Eurobot is available, with soundtrack in 6 different languages. It can be downloaded [1].

Beside Eurobot, several robot competitions exist at world level, in particular FIRST, initiated in America and Robocup, initiated in Asia [2-5]. While they all feature a variety of leagues, their strong points are different. FIRST is probably the best for the very young (“Lego league”; with key material fully available and documented on the market, socially oriented educational aspects, and in particular also non-technical aspects in selection rules). Robocup is probably, globally, best profiled for academic research (“senior leagues”) and includes “applied, useful – no game” leagues (“rescue”, “at-Home”). While also bringing excellent contributions in these fields, Eurobot is unique in the mostly educational challenges it successfully defines, years after years, resulting, for students and fans, in the design of hundreds of mostly dextrous and agile robots, always original, and matching novel, specific rules.

After 10 years of experience, time has come to put into perspective some of the outcomes of Eurobot initiative. We do it below from our own perspective, but we are convinced that many teams would share most of our conclusions.

The paper is organized as follows. Section 2 first presents our platform family for Eurobot. Then Sections 3 and 4 group remarks and major findings about Eurobot, as perceived from respectively two view angles : first Education and research , and second Science and technology. In most case, there is an interesting tension between, schematically speaking, two opposite approaches, which each have their own merits.

2 Eurobot and ARY platform

Eurobot has brought a totally new possibility for students, engineers and members of technical clubs to work on a well defined problem, to compare their solutions to the solutions of peers, and finally to disseminate their results to a large public. In some aspects a very large autonomy is left to the participants for them to solve the problems (every year a different task is defined), and on the other hand some rules are strictly defined, including requirements for total autonomy, restricted volume, short play time (90 seconds), random components of task definitions (in addition to the unknown strategy of opposing robots on the common playground), effectiveness in real world (ego motion, motion of objects), and precise time when robots have to be ready for competitions.

Eurobot strict requirements induce serious challenges in scientific and technical terms. The two main ones that appeared, when we started addressing the design of such autonomous robots, are the following ones:

- 1. necessity to design a fully autonomous mechatronic system, active in physical space, specifically matching task properties: e.g. grasping, storing and shooting squash balls (ca. 3 cm diameter), or tennis balls, or rhythmic gymnastic balls (15 cm dia.), or soda PET bottles, or 1.5V batteries, or wooden pins with metallic inserts, etc.;
- 2. necessity to design a powerful programming environment, which allows for very fast adaptation of configurations and strategies.

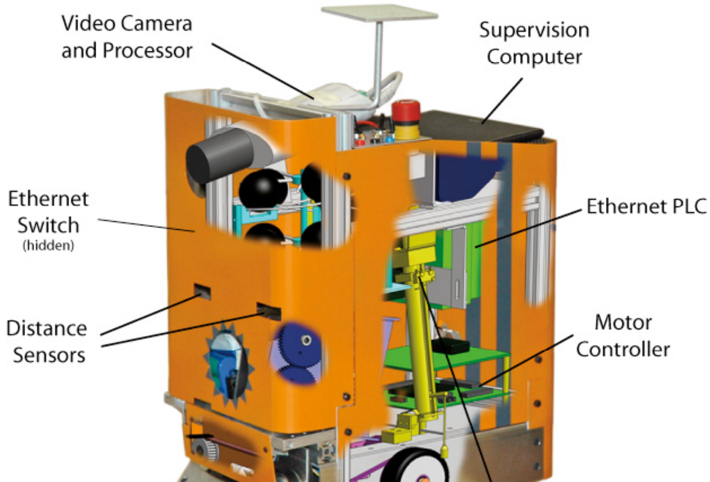

Fig. 2. "Dude", with the help of another smaller robot, "Walter", was engaged in the bowling tournament of Eurobot. It could pick, store, and shoot squash balls in order to make its skittles fall; as well as to restore in an upright position the possibly fallen skittles of opponents.

2.1 Necessity to design novel, specific, mechatronic systems in physical space

For our family of ARY robots (e.g. [6-9] and Fig.2-4) , we have attempted to ease the necessity to design novel, specific, mechatronic systems in physical space by using in as much as possible COTS elements. But unfortunately very little exists in mechanical terms, beyond bolts and screws, linear frame bars, and motor-gear-sensor assemblies (e.g. Fig. 2-4). What makes things worse is the necessity to pack a maximum of mechanical functions and components in a restricted volume.

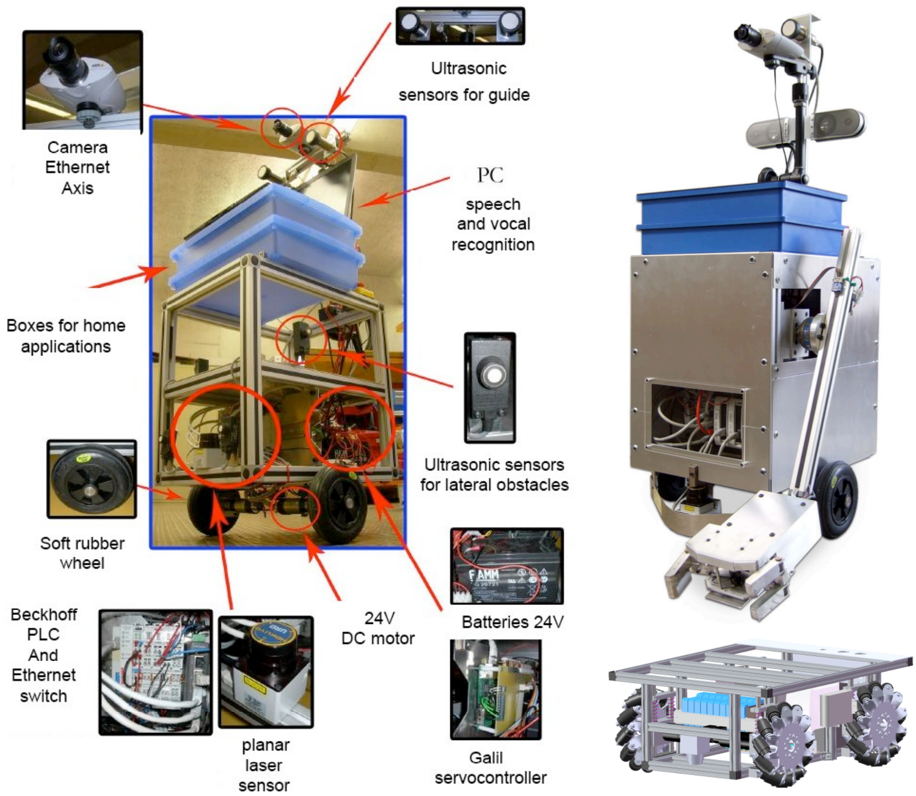

Fig. 3. "RH2-Y"(left) and "RH3-Y"(right, with new locomotion platform in development), have been designed for "at-Home" applications and inherit many of the developments made for Eurobot. Additions: vocal dialogue, laser scanner (visible above), arm, hand, etc.

Many years ago, in a different context [10], we used to win a robot competition, essentially in two ways, similarly successful: one was using a camera, a PC and an industrial robot, and the other one, provocatively for an AI context, was successful without any sensor nor digital processor! But in the latter case, it appeared that in a technique very similar to usual programming, where a few language primitives are progressively structured in larger, integrated procedures, we had to a large extent, a mechanical kit whereby small standard elements where joined in structures larger and larger, each ensuring by mechanical constraints that the required functions be enforced (moving, sorting, reorienting, lining-up, etc.).

In Eurobot context, constraints of limited volume and very large changes in tasks-to-be-done really force teams, every year, to innovate, in order to successfully design new, efficient, mechatronic action systems.

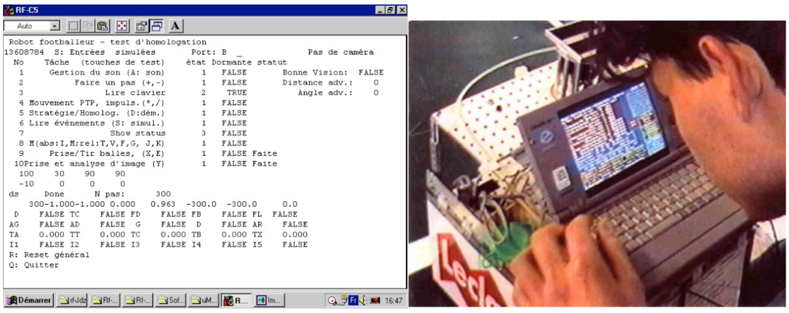

Fig. 4. First versions of Piaget environment, on Diego[3], 1998.

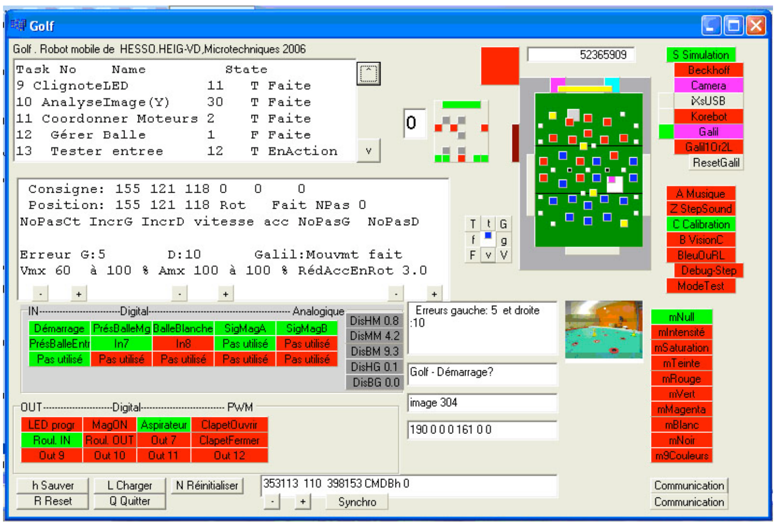

Fig. 5. Later versions of Piaget programming and control environment, on Dark-18, 2006.

2.2 Necessity to design a powerful programming environment

Eurobot participants face with a very high degree of uncertainty in the improvement of their machines. Therefore many very fast changes are necessary during development and competition phases in order to reach a certain level of excellence. In addition, the development of the system is made in such a way that typically, very little time is available after the moment when robots are fully integrated. Therefore, extensive possibilities in terms of simulation are helpful, so as to allow for parallel engineering.

It has been clear for us from the beginning that for sophisticated programming, extensive simulation and man-machine interaction, PC technology was a good choice. For parallel programming and real-time control however, common tools were not appropriate. We developed our Piaget environment, including a very fast multitasking kernel and an application-oriented language.

The second year, Piaget was adapted to include industrial robot types of instructions (a subset VAL instructions: move, appro, signal, inverse, tool, etc.). The reason was that when beginning with a new task (game), it is not known yet which sensors and actuators will be used, and which elements of strategy will be designed; thus to be able to handle programming as for a standard robot is an advantage during early phases.

As can be guessed from Fig. 4 to 6, Piaget environment has remained remarkably constant, even though at implementation level, major changes have occurred, in particular moving from Pascal to C++, from DOS to Windows, and more importantly yet, from a conveniently centralized structure – the PC- with very high reactivity possibilities via standard parallel interface (ca. 1 microsecond per transaction) to a strongly distributed architecture, featuring Ethernet-TCP-IP communication standard (ca 0.1 s per transaction), and, along with PC supervision, a significant number of specialized subunits: motion controllers, PLC, camera with embedded low-level processing unit, etc.

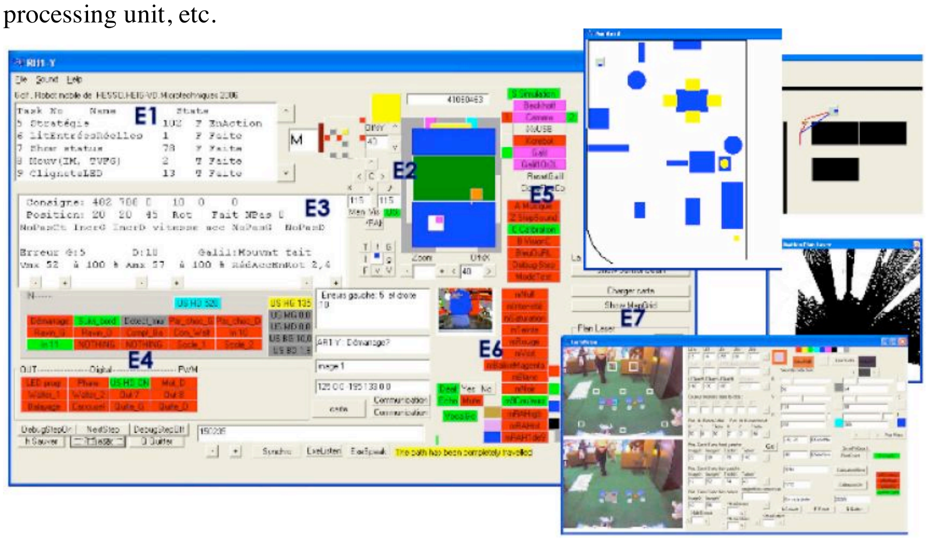

Fig. 6. Current version of Piaget programming and control environment, on RH2-Y, 2007. In addition to main screen, 4 windows show a map of Atlanta "Robocup-at-Home" facility, virtual trajectories with detection of possible collisions, laser measured distances, and some vision processes for "CopyCat" test.

The main control panel allows to monitor I/O states, progress in all parallel tasks, scenes, distances, and much more. It can also with immediate effect change I/O states, especially in simulation mode for what concerns inputs, trigger various sorts of motions; a third category of actions allows for reconfiguration and parametrization of the robot, including the possibility to store the new configuration for later runs.

3 Education and Research

For our experience, Eurobot has been useful both for education and research, and importantly, it has also proven effective in helping bridge the gap between education and research.

3.1 Eurobot and education

Before Eurobot, our courses were taught with a focus on the areas of robotics generally found essential. And experiments in the lab were offered, with less short-term incentive than when competition context started to be considered.

Eurobot has forced a shake-up of course content, so as to consider in priority urgent items. After such a first visit of course material, a second path was done, filling-up information which had been bypassed in the first turn. More importantly, gaps in the classical training became apparent, and called for immediate action in terms of extended training/coaching, thereby fostering transfer from research to education.

From 1998 to 2005, in our school, like in previous years, there was enough time allocated to robotics and automation, for a curriculum in microtechnology, so that 2 periods per week could be used to perform coaching. The advantage of this very good approach is, while a lot of initiative is anyway left to the students, that a careful coordination is possible with regular course and lab contents, in close synergy with curriculum requirements; and reciprocally that students understand that such an experience is a necessity for their professional preparation. (Now, unfortunately, curriculum revisions in line with European bachelor and master recommendations have reduced the total amount of periods allocated to the program, forcing Eurobot preparation to be performed separately, in independent clubs.)

An unexpected effect of Eurobot initiative has been to trigger interest in the whole educational line for our region, reaching, beyond universities and even technical and vocational schools, classes of secondary and even primary schools. Of course this has happened with an adaptation of content, of roles for the students or even as themes and approaches (FIRST Lego Leagues, or even simply – but to a large scale – local school Cups). This has brought a lot of benefits in introducing the fun of science and technique to the youth and even to carry science more broadly, in the society at large.

3.2 Research and innovation

Eurobot competitions call for “systems”. No matter how advanced the state of the art is in many areas, there are always finally one or several components that turn critical.

They represent the limiting factors, and there is no way for improvement but to perform effective research on those items.

As mentioned above, a key requirement was to design a flexible, multi-tasking/multi-agent, real-time kernel. If a team wishes to reach the top half of the Eurobot score list, it is absolutely impossible to program the robot with a single programming thread; or to ignore (real) time-based constraints. We have started from an old approach made famous by Texas Instruments in the early 80's: context-switching. With progresses in microelectronics (high frequencies, large memory sizes) and in computer engineering (e.g. cache-memory mechanism), this has allowed us to have today, typically, our PC-based agents active, in average, every 2 microseconds; they start-work-and-leave within time slots of an average of about 100 nanoseconds. And this integrates the average time spent in all other Windows-tasks! With such a concept, priorities or interrupts do not need to be considered; polling is effective and makes task/agent synchronization a very simple thing to implement.

In research, mostly two main challenges have been identified and solved beyond previous state of the art: quantitative assessment of cognitive properties (in automated systems: quantitative cognitics), and ensuring a safe relative agility of decision-making resources, for all closed-control paths (stability of multi-agent systems) [e.g. 11].

4 Science and Technology

This section includes a number of focused remarks on science and technology aspects, as they often emerged during the first 10 years of Eurobot experience. They relate below successively to AI, control loops, architecture and motor technology.

4.1 AI versus software engineering

Our laboratory, in the scope of industrial robotics and automation, has always been following progresses in artificial intelligence (and sometimes making contributions in this area as well).

Experiments have been fully programmed for approaches typical of AI such as below:

- Genetic algorithm and natural selection process for control design (successful pole control, in Delphi and C++)
- Neural networks for successful pole control, in Pascal and C++ (using Hopfield model, Hebb law, and also Backpropagation (C++ only))
- Fuzzy logic for successful pole control, in Pascal and C
- Memoryless animats (re. R. Brooks concept) for labyrinth search (deterministic) and collision-free navigation in labs (stochastic approach)
- Navigation algorithms with learning for collision-free navifgation in labs.
- Tree search with forward and backward chaining, breadth-first and depth-first approach (Nokia-type puzzle, C++)
- Eliza (Pascal and C++)

- robotic) handling of cubes on piles, using “expert-systems”, i.e. declarative programming in “PC-Plus” (special environment for expert-systems, implemented in Lisp), , Prolog, Lisp, Basic and Pascal (re. “beyond prototyping”).

Our main contribution can probably found in quantitative cognitics. By this technique, it is obvious that for demanding cognitive tasks such those typical of Eurobot competitions, all above AI methods cannot be used (at least not as the main driver; but of course it could always be considered for very minor, ancillary functions) and on the contrary software engineering (or possibly logic-hardware based engineering; or even embedded approaches) provides far better possibilities.

4.2 Active or “passive” control loops

Feedback loops may have some value, even if not explicit and “digital”, but simply embedded in physical systems .

For example the wheels of railway trains are commonly kept between rails without sensor nor computer, but just as a result of lateral forces and material elasticity (F=kx; F=ma).

Similarly, a stepper motor will naturally deliver a corrective torque if disturbed from its nominal position, just as a DC-motor with encoder and digital controller would do.

(Still, slamming a robot against Eurobot table sides, as is often seen, though effective, is not so elegant for pose calibration.)

4.3 Parallel port or Ethernet TCP-IP architectures

Parallel ports are gone, and distributed architecture has novel merits.

In early times, parallel ports were standard, even on very compact computers, and the OS allowed for fast and direct access to physical addresses, for user applications. Along with a lean OS, this could allow for a “total” programming of applications on the PC.

Today, the parallel port is just not available, even not as an option on small systems. But Ethernet can be found everywhere. Similarly OS’s have evolved, offering on demand a really huge amount of interesting options.

The cost of this new architecture is the poor reactivity of PC at outside events, in standard configurations; therefore a multi-agent architecture cannot be avoided, with specialised resources for serving all fast processes (in particular, the servo control of small electric motors).

4.4 CC or brushless versus stepper motor

A question typically arises when starting a new robot : which motor technology would be the best for locomotion.

Schematically speaking, CC or brushless motors can be considered here to be equivalent, and will be referred to as servoed motors. (Compared to CC motors, brushless motors have the potential of turning faster and lasting longer, but they are somewhat more complicated then DC motors to handle and should in principle not be required for Eurobot competitions).

Stepper motors have a kind of Boolean behavior: if by design, steps cannot be forced (typically, the wheel slips on the floor before the motor misses steps), then they may be viewed as “perfect”. Otherwise, they cannot be used.

On the other hand, servoed motors seem always to work. But depending on control quality, best performances may be hard to reach, and in fact performances can degrade significantly, which may translate in important errors on trajectories.

On former designs, we had typically stepper motors, with excellent performances and very simple, 2 bit interface with PC. Since we have switched to the distributed architecture, we have been using DC motors, with dedicated controllers, interfaced with Ethernet TCP/IP. Servocontrol and trajectory laws (trapezoid speed curve) are specified by the supervisory PC, and implemented locally.

4.5 Other remarks

Other remarks may be useful.

Top-down or bottom-up approach (free-wheel sensors) for location estimation. We have nearly always used a top-down approach for the ego localization of robots. Typically a 1 cm accuracy, i.e. on the order of 1 0/00 could be achieved through a match. Recalibration, with vision or laser sensor, have often been implemented, and sometimes been practiced during training phases, but it is hard to be as accurate, and it takes time and risks. We have also tried the use of free-wheels, which may be necessary if motorized wheels slip significantly because of highly dynamic behavior, but in our case where motions are carefully parametrized, integration of those 2 signals to estimate 3 coordinates in the plane is comparatively not accurate.

About quantitative cognitics. Cognitics is the science and techniques of automated cognition. More generally, we have found necessary to introduce a proper metric system for cognitive processes. Take an analogy: can typical humans jump over a wall? Well, it depends on the height of the wall. Yes for 0.5 meter; No for a 5m high wall. For cognitive tasks too, quantities matter; e.g. how complex is the task? How much knowledge does it require? etc. We have therefore introduced units and mathematical equations for cognition, based on information and time, “bit” and “second” units; in particular “bit” for complexity, “lin” for knowledge, “lin/s” for expertise and learning (e.g. [11, 12]).

Continuous versus PTP motion. Newcomers typically have the feeling that motions should be ordered as preferably “long” and continuous, rather than “point-to-point”, and incremental. In fact it is possible, for the previous case, to have parallel agents watching for changes in circumstances and possibly requiring emergency stops and changes in motion plans. But in practice, this requires a good synchronization of several processes, which is never obvious, and more importantly, this makes the programming of "next" actions extremely difficult: by this token, a very large variety of situations may occur, and to define practically what has to be done next, for each and all possible situations, may just turn out to be too difficult to be done.

Holonomic versus non-holonomic platform. Moving in the plane requires 3 actuators to allow for immediate motions. In our case, we usually use 2 motors and active wheels (very common approach), and "Amiguet '98" method to successively implement 3 virtual joints (θ_1, d_2, θ_3)), i.e., typically, straight-line motions; or sometimes we also implement circular interpolation (this is especially useful for leaving from very confined locations, i.e. to somehow simultaneously translate and turn). If we relax the constraints for simplicity, volume, and cost, the holonomic approach may be interesting. We are now in the process of designing a 4-motor-Mecchanum platform for our domestic robot, with the idea that this may help not only for locomotion but also for handling: the robot arm may so inherit from the 3 dof of the locomotion resource.

Other interesting elements of experience could also be gained in other areas, notably in the following domains: PLC versus PC or servocontrollers; PC versus Integrated PC; stereovision versus laser distance estimation; single or multiple ultrasonic sensors (issues relating to perturbations); similarity of information quantities in B&W versus color versus 3D images; Amiguet's method versus splines, clothoids and other approaches (re. Serpentine) for trajectory generation; dual need of teaching basics and offering tutorials versus incrementally communicating newest R&D results. There is however no more place here for discussing those points, and the reader is invited to refer to our website or other publications to know more about them.

5. Conclusion

After 10 years of participation to Eurobot, time has come to sketch a summary of knowledge and expertise gained so far. Eurobot has brought on the world scene, typically for students in engineering schools and fans of technology, a unique opportunity to design autonomous mobile robots, most often with excellent agility and dexterity properties. Eurobot typically sets two major challenges: design a fully autonomous mechatronic system; and design a powerful programming and control environment for very numerous and fast adaptations. Here mechatronic systems require a lot of original design, because of the requirements, yearly renewed, and because of the restricted volume allowed. As programming environment, we have created Piaget, which is very simple, effective, and quite constant for the final user, even though its implementation has changed a lot through the years, in terms of implementation language, operating system, and control architecture. Education and research have both benefited from Eurobot initiative. For education, in addition to the positive effect on students directly involved, Eurobot has contributed to a much larger impact on society, notably at schools and in the training of the youth. In research, we have in particular found that improvements in quantitative cognitics, and considerations about the relative agility of agents in control loops were important contributions to be fostered. The paper still brings a number of remarks emerging from a decade of Eurobot experience.

In terms of points and rank, our goal has always been to pass the homologation test, so as to be able to actively participate in the event, with a probability of 95%. It turns out that until now the team has constantly had a 100% success rate at homologation test, both at national and European levels. In addition, several cups have been obtained.

In terms of organization, while in the past advertising, with the involvement of television channels have been a good support, for the future other paths should be explored, particularly in the direction of public support for edutainment and public awareness in science and technology, along with the classical education resources and private support.

References

1. *Description of Eurobot competitions*, 6 minutes, on DVD, Eurobot, oct. 2004. For download : http://lara.heig-vd.ch/publications/VideoWMProdEurobot2004SixMin/En/VTS011.VOB.zip ; or visit [8]

2.-Jean-Daniel Dessimoz, Pierre-François Gauthey and Carl Kjeldsen, "Interest of Ludic Competitions for Robotic Education and Research",Workshop on Educational Robotics 2006, org. University of Catania, Eurobot, and IEEE, with support of the European Commission-Directorate General for Research, Science and Society Program , Acireale (Catania), Italy, June 1st, 2006, pp10.

3. www.eurobot.org Eurobot official website

4. www.robocup.org , Robocup official website ; also : RobocupAtHome League: Goals and Modalities: available on: http://www.robocupathome.org (2007).

5. www.usfirst.org , FIRST de facto official website

6 Nicolas Uebelhart, Florian Glardon and Pierre-François Gauthey, "Lomu, an Autonomous Mobile Robot with Robust Architecture and Components", « DARH-2005 - 1st International Conference on Dextrous Autonomous Robots and Humanoids», with sponsorship Eurobot, IEEE, CLAWAR, and CTI, HESSO-HEIG (West Switzerland University of Applied Sciences), Yverdon-les-Bains, Switzerland, May 19-22, 2005

7 P. Maurer and M. Gagnebin, "Advanced control structure for the autonomous mobile robot Lodur", «DARH-2005-1st International Conference on Dextrous Autonomous Robots and Humanoids», spons. Eurobot, IEEE, CLAWAR, CTI, HESSO-HEIG (West Switzerland University of Applied Sciences), Yverdon-les-Bains, Switzerland, May 19-22, 2005.

8. LaRA robots and Eurobot DVD : http://larae.populus.org/rub/4

9. Jean-Daniel Dessimoz, Pierre-François Gauthey, "RH2-Y – Toward A Cooperating Robot for Home Applications", Robocup-at-Home League, Proceedings Robocup07 Symposium and World Competition, Georgia Tech, Atlanta, USA, 30 June-10 July 2007.

10 J.-D. Dessimoz, A. Beran, S. Ernst, O. Olmo and L. Venries, "Two Antinomical Methods For Automatically Solving Assembly Tasks; Case of Picking and Stacking Film Cans", Proc. 1994 Latsis Conference: From Perception to Action (Per'Ac), at EPFL, Lausanne, Switzerland, Sept.4-9, publ. IEEE Comp. Soc., New York, 1994 (winner of Robot Contest).

11. Dessimoz, J.-D., et al.: Ontology for Cognitics, Closed-Loop Agility Constraint, and Case Study – a Mobile Robot with Industrial-Grade Components. *Proc. 5[nd] IEEE Intern. Conf. on Industrial Informatics INDIN'06*, August 2006, Singapore (2006)
12. Jean-Daniel Dessimoz, "About the Necessary Move from Cognitics to Ethics; Additional Definitions, and Contributions to Metrics in MCS ",« DARH-2005 - 1st International Conference on Dextrous Autonomous Robots and Humanoids», with sponsorship Eurobot, IEEE, CLAWAR, and CTI, HESSO-HEIG (West Switzerland University of Applied Sciences), Yverdon-les-Bains, Switzerland, May 19-22, 2005. (re: www.darh-2005.org)