Res Eng Design
DOI 10.1007/s00163-014-0171-y

ORIGINAL PAPER

Designing winning robots by careful design of their development process

Eli Kolberg · Yoram Reich · Ilya Levin

Received: 26 February 2014 / Revised: 27 February 2014 / Accepted: 28 February 2014

Abstract We present a comprehensive robot development process and its evaluation. We designed this process in the context of a robotics course in high schools. The motivation for designing this new process was improving the robustness and reliability of robots developed by students and preparing students for becoming better designers. The newly designed process proved to be highly successful in designing top quality robots. In the process design, we explored and adapted existing design tools and methods to the specific designers, the nature of the product, the environment, the product needs, and the design context goals. At the end of this thorough design, we selected a synergetic integration of six tools and methods to compose the new comprehensive development process for this product context: conceptual design, fault-tolerant design, atomic requirements, fuzzy logic for control, creative thinking, and microprogramming-based design. The design skills of the students that learned the design process and the performance of robots they designed and participated in an international robotics contest were examined. The high school teams that studied the proposed process won the first places in an international contest. The robots developed by the students had better performance than robots built by engineers and faculty teams. Professional experts rated the robots' designs as excellent. The students that studied the process demonstrated high level of diverse design skills including creativity and design management capabilities. Additionally, they improved their science subject grades and their attitude toward engineering. Both the results obtained by the study and the authors' experience in teaching robotics demonstrate that the proposed robot development process could be taught successfully in high school and that it leads to superior robotic products. Our experience also indicates that this process could serve industry design by improving the robustness of robots operating in uncertain environments and supporting fast change management practices.

Keywords Robots development process · Robot design process · Design education · Research methodology · Mechatronics design process

Small parts of this paper appeared previously in different publications by the authors and are included in the reference list. However, this is the first and only comprehensive description of the study goals, design, research methodology, results and conclusions.

E. Kolberg
Faculty of Engineering, Bar-Ilan University,
52900 Ramat Gan, Israel
e-mail: eli.kolberg@biu.ac.il

Y. Reich (✉)
Faculty of Engineering,
Tel-Aviv University, Tel Aviv, Israel
e-mail: yoram@eng.tau.ac.il

I. Levin
School of Education,
Tel-Aviv University, Tel Aviv, Israel
e-mail: ilia1@post.tau.ac.il

1 Introduction

Robots in all their forms are products that increasingly penetrate into diverse applications and markets. Parallel to this trend and in order to facilitate it, autonomous robots are expected to be more reliable, especially when used in home environments as personal assistants or in hospitals as surgical aids. As all products, they obviously need to be good quality products that address well their customer needs. Given the complexity of robots as products and the complexity of their design environment (e.g., they are designed by multidisciplinary teams using diverse tools), designers need some support in structuring their design process and selecting their tools and methods. However, a review of studies and books on the design of robotic or mechatronic systems reveals that they do not really discuss designing robots, but rather, they discuss the technology related to robots: the behavior of components such as sensors and manipulators or control. No reference provides methods or guidance in interpreting a design brief or a problem situation into a sequence of steps that leads to robust final product (Dertien 2006; Jones et al. 1999; Tummala et al. 2002). This omission is reflected, for example, in the formation of a European Consortium to remedy this situation because "even after 50 years of robotics development and research the process of developing a new robot and its applications has more similarities with ingenious engineering or designing a piece of artwork than with a structured and well-defined process. This holds particularly true for advanced service robot systems[1] (BRICS 2011)." The BRICS Web site[2] further notes that in 2009, Google Scholar search produced 26 results for "robot development process"; in January 11, 2012, such search produced 60 results, but none of which really describes a development process as it is commonly understood by product development professionals.

Is the above an acceptable situation or could it be improved? We contend that many failures of robotic or mechatronic products could be attributed to design errors caused by poor development processes, whether lacking the use of appropriate design methods or failures of development process management. The falling of a Segway when its battery runs out of power leading to halting its control is a simple example of bad process management. A detailed analysis of robot failures also reveals that some failures could have been avoided with better user-centered design processes, and other failures could have been avoided by the use of better design processes (Carlson and Murphy 2005).

Our goal in the present research was to develop a new comprehensive robot development process (CORDEP) (Reich et al. 2005) and prove that its use benefits designers. Such goal immediately raises serious methodological questions regarding which proof would be acceptable to justify our claim. The common methodology would be to demonstrate such method on a case study. We decided that such demonstration might be limited and potentially subjected to scrutiny. Therefore, we wanted to subject the method to a large-scale testing where statistical analysis would be possible. However, such decision is impossible to implement in real design practice as no real design is ever executed twice, not to mention enough times to constitute a sample for a serious statistical test.

Consequently, we decided to seek an environment in which reasonably serious design is carried out by multiple teams over an extended time that could be the basis for a large-scale comparative test. We selected an educational setting of a robotics course, in which senior students majoring in science from four high schools build autonomous mobile robots for participation in an international robotics contest (Verner et al. 1997). The contest participants are engineers, university students, and faculty besides high school students and as such can serve as a good test field of the robot development process compared to real practice. Our experience in several years of conducting this course was replicating the failure patterns in commercial-level robotic products we mentioned before. In our case, it was clear that the lack of using proper design methods in a well-managed process led to inferior designs that suffered from recurring problems (Kolberg et al. 2003). A confirmation of our hypothesis—that using appropriate design process and methods leads to designing better products—would be obtained by submitting it to an appropriate test in several high schools. Of course, even with such testing, there is still a gap between the educational and an industrial setting. We hope that readers will be able to foresee the value of the results also to industry, and we leave such demonstration to another study.

The remainder of this paper describes the design of the design process and its testing. Section 2 describes the overall research methodology of this study. Section 3 describes the design of the robots' design process. Section 4 describes the particular design process developed in this research. The results of the study are given in Sect. 5, and Sect. 6 includes some recent consequences. Section 7 discusses the results, and Sect. 8 concludes the paper.

2 Research methodology

The robot development process is in the focus of our research. Due to the need to study it well, we translated this to developing a robot development process for an educational setting. Consequently, we now have another issue to address: If we are designing a design development process for robots, we need to design it according to some requirements. What if the requirements of the educational setting are different from those of an industrial setting? In such case, the resulting design would be different. Indeed, we contend that a design development process should fit its context and that there is no one single best general design process or method (Reich 2010). Are we contradicting ourselves or defeating a priori our goal?

[1] See http://www.best-of-robotics.org/brics-in-a-nutshell, accessed 8.8.2013.

[2] See http://www.best-of-robotics.org/brics-in-a-nutshell/robot-development-process, accessed 8.8.2013.

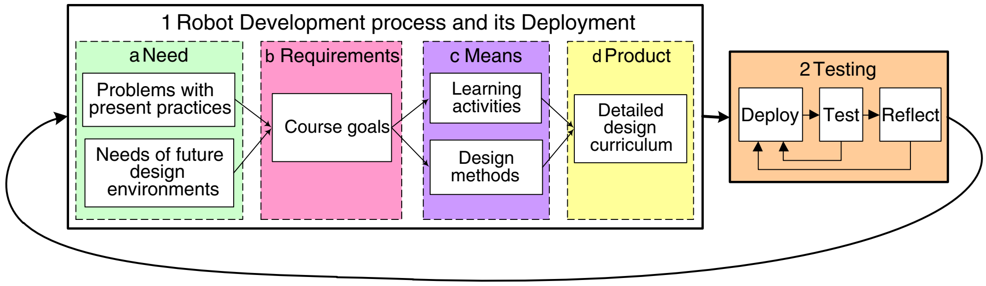


Fig. 1 Road map for designing design processes

We contend that in spite of this difficulty, we can still execute this study as planned. We will demonstrate how a careful design of a design development process is done and how the result—a robot development process—is used successfully by high school designers. We then argue that in an industrial setting, the design of the development process would be executed again and will lead to results that are based on the same tools with some variations to accommodate for the new context.

Figure 1 describes the road map of this study divided into two main steps. The first step is devoted to developing the robot development process and its training procedure (as teaching material for high school students). The first step is composed of 4 tasks. First, we define the needs and requirements of the development process (tasks 1a–1b in the figure) and subsequently design the development process and its deployment (tasks 1c–1d).

Since the first step involves developing an educational course, we have to delve into the educational area and not only remain at the level of the robot development process. This will allow us to contribute also to the field of education. There have been many studies on the design of curriculum in education (e.g., Barrows 1985; Clark 1997; Crawley et al. 2007; Diamond 1998); however, we are not aware of a large-scale study on teaching robot design that was tested in a controlled experiment and produced conclusive results as the present study.

The second step of this research tests the development process by implementing the course and studying the robots that are developed in the class. The results of the test are an important resource for study and redesigning the robot development process. One of the primary means of the testing is participating in an international robotics competition. Robotics contests as appears in Ahlgren and Verner 2002; Martin 1994; Kitano 1998; Sklar and Eguchi 2005; Kolberg et al. 2005; Reich et al. 2005 serve as a tool for comparison between different robots; in addition to such comparisons, competitions such as the DARPA Robotics Challenges[3] also drive the development of new technologies. As the competition participants are also engineers and university students, the test has high degree of significance. Altogether, we tested the results of the robot development process in three consecutive years (Reich et al. 2006) and report on some longer-term results.

[3] See http://www.darpa.mil/Our_Work/TTO/Programs/DARPA_Robotics_Challenge.aspx, accessed 5. 5. 2013.

3 Designing the robot development process

This section describes the proposed design of the robot development process (CORDEP). This recursive nature of designing (here, of a development/design process) is central to design since anything we do is a result of a design including the methods or the processes we use. Acknowledging this situation is the first step to understand that there is no single better design process but diverse processes or methods that each fits better a particular context (Reich 2010).

3.1 Design methods used to design the robot development method

We first briefly describe the design methods we used to design the development process. Additional information about them could be found in various references.

Function-means tree (Hubka 1974) is a visual representation of decomposing the function of a product into subfunctions that are realized by some means. Each mean may, in turn, call for further subfunctions to be realized by other means.

Failure analysis deals with investigating past failures in order to determine their causes and identifying methods that could remedy them.

Quality function deployment (*QFD*) (Akao and Mazur 2003) is an established collection of conceptual design tools. The purpose of these tools is to guide product developers from abstract ideas, problems, or needs through information gathering, to find customer requirements, and to translate them to engineering information that could drive developing good product concepts. The most famous tool in the suite is the *house of quality* (*HOQ*), and its main goal is to translate between customer requirements and engineering properties. Nevertheless, the HOQ could be used to translate between any two sets of properties as well as for selecting between alternatives if used properly.

Another popular conceptual design tool is *Pugh concept convergence method* (Pugh 1991). This tool qualitatively compares each concept relative to a reference or datum concept for each criterion. After rating, a process of improving the best concepts is exercised, leading to a new set of candidates that is again rated against a datum. This process usually converges after few iterations (Frey et al. 2009).

Analytic Hierarchy Process (AHP) (Saaty 1980) is a method for prioritizing alternatives for addressing a set of criteria. The criteria could be organized hierarchically. AHP exercises pairwise comparisons, leading to a ratio scale that represents how much one element dominates another with respect to a given criterion. As judgments may be inconsistent, AHP includes a way to detect inconsistencies and improve the judgments in an attempt to improve consistency.

Within conceptual design, one of the tools to prevent proactively future failures is *Failure Mode and Effects Analysis (FMEA)* (Rhee and Ishii 2003). FMEA is composed of numerous steps (Crow 2002) including describing the product and its functions; identifying potential failure modes and the effects of these modes, as well as their causes; ranking the severity of the effects; and choosing those modes that we would like to avoid by modifying the design.

3.2 Need and requirements articulation

After several years of teaching courses on robot design with conventional design process, which include the following steps: establishing the need for a specific product, defining the product requirements, and creating and choosing the leading solution, detailed design, and system building and testing (Ben-Hanan and Reichsfeld 2008), it was apparent that recurring problems were manifested (Fig. 1a). Failure analysis of previous course problems pointed out the following issues:

1. Numerous times, teachers, and course participants, designed things that did not fit with engineering design know-how, which, in turn, caused robot faults and teachers and students disappointments.
2. In that time, there was no standard or orderly process that directed students' attention to what is customary in developing an engineering product in general and a robotic product specifically.
3. Each year, schools purchased expensive equipment, and sizeable part of it had never been used due to different reasons, among them, the lack of appropriate design methodology.
4. Participants had no clear understanding of the relations between science and engineering.

In addition, three other general observations emerged:

1. The students have never faced projects with such high level of complexity that resembles a real-world project.
2. The students have not experienced real teamwork, such as when engaging in a large-scale mechatronics project.

As an educational course, we wanted also to include in the training some preparation for working in future design environments which reflect a world with accelerated speed of change and an increase in the complexity and multidisciplinarity of products. Of course, the nature of the product—an autonomous robot—makes a difference in determining the development process and its training.

The broad needs were translated to course goals (Fig. 1b) using a simple function-means tree. It was sufficient for our purpose although more elaborate methods could have been exercised (e.g., QFD). The following were the derived course goals:

1. Acquiring technical knowledge;
2. Acquiring system thinking;
3. Improving skills of problem solving, decision making, and self-learning;
4. Developing critical and creative thinking abilities;
5. Experiencing development of an autonomous mobile robot, with time and budget restrictions;
6. Developing teamwork skills;
7. Improving students' design skills; and
8. Improving students' perception of technology.

3.3 Determining learning activities and design methods

Having established the course goals or the requirements, we need to create a concept that will then be detailed to a full design. The concept of the course includes the development process to be taught, composed of a sequence of design methods, and the learning activities that will train students with the process as well as teach them all the required knowledge and skills (Fig. 1c). Since the focus of this paper is the design of the development process, we do not elaborate on the learning activities.

We decided to teach students sufficient design methods that would allow them to design and build robots of high quality but without overwhelming them. Design methods are seldom taught in high schools. Moreover, despite the importance of the design methods, many universities do not teach them either. Majority of universities focuses on analytic rather than synthetic skills related to design methods. Clearly, there are exceptions, but even if they are exercised in 100 (e.g., through initiatives such as CDIO; (Crawley et al. 2007)) or even in 1,000 universities, this number is still small.[4]

Our reasons for teaching design methods stem from the course's second to seventh objectives. Design methods serve as guidelines that help students (as well as designers in real design projects) focus on the critical features when developing any engineering product. In addition, design methods glue technology to science. They help students realize the relations between the different science subjects learned at school and between science and the engineering work in robotics. These are central to understanding robotics as a discipline.

Further support for systematic teaching of design methods arises from feedback obtained from previous courses on the subject. One observation was that ignorance of design methods prevents effective use of expensive equipment purchased to support new technology and science-related courses. Another observation was that lack of knowledge about design methods led to numerous occasions in which teams designed robots that violated simple engineering practice, resulting in quick robot failures. While product success is not mandatory for course success, these easily avoidable failures led to students and teacher disappointments, which are undesired. The logic for designing CORDEP and selecting the design methods is presented in Fig. 2 and elaborated in subsequent sections.

3.4 Design methods and their initial assessment

We collected a set focal topics and design methods that are general (Ullman 1992; Dieter 2000) as well as relevant to the context and sorted them, see (1) in Fig. 2:

1. Product architecture design (Dahmus et al. 2001; Hubka 1974; Morag 2003; Ulrich 1995) determines the arrangement of the physical elements of the product in order to carry out its required functions. This is important for any complex system, so it is kept for further analysis.
2. Configuration design (Franke 1998; Myung and Han 2001) deals with how to assemble all the designed components into the complete product and maintain its structure. As the robot has components that have to be assembled and their structure maintained and managed well throughout the project, we keep it to the next stage.

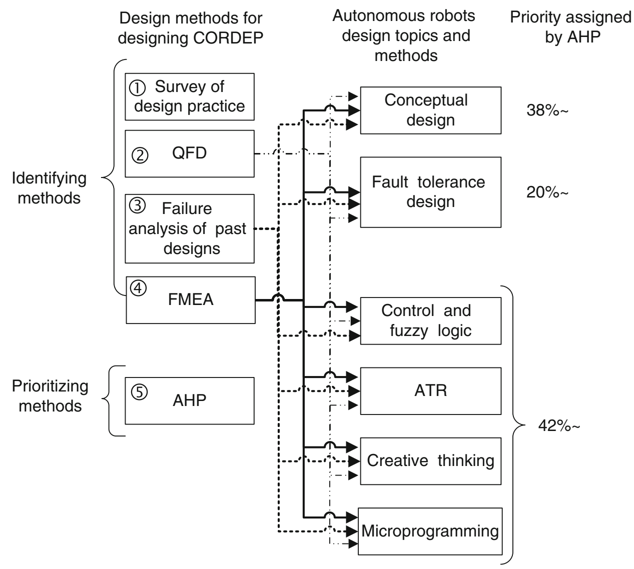


Fig. 2 Course design topics and leading to robotics design method

3. Parametric design (Myung and Han 2001; Zhang et al. 1995) identifies the attributes of parts in a design configuration that become the design variables for detailed design. The objective is setting values for the design variables that will produce the best possible design considering all product requirements. As optimal performance is always desired, we kept parametric design to further analysis.
4. Conceptual design (CD) (Reich 2008; Ullman 1992; Wang et al. 2002; Li 1996; Cohen et al. 1992) is one of the two most critical stages in product development. Tools used in this stage, to organize information, create a high-level view of the product and identify important factors that are important. The insight created by these tools could be used also to divide the work between team members and complete the project on time. Hence, we kept these tools for further analysis.
5. Concurrent design (Prasad 1996) deals with cross-functional design team, where skills from the functional areas are embedded in the team. This allows for parallel product design. In our case of small teams, where there are no design skills differences among team members, it did not seem worthwhile to consider this approach further.
6. Atomic requirements (ATRs)-based design (Levin et al. 2004b; Salzer and Levin 2004; Galster et al. 2007) divides the set of initial requirements into very basic and thus simple for understanding requirements. It helps to identify unnecessary, overlapping, or conflicting requirements, isolate bug areas, and make

[4] There are more than 20,000 universities in the world. www.webometrics.info/methodology.html, accessed 2.6.12. Updated count (1. 1. 14) lists over 21,000 and estimated number of high education institutes of about 40,000.

clear what is to be done to implement the requirements. In the debugging mode and problem solving, each requirement can be tested easily and separately. It is an efficient communication tool between people of different backgrounds. The ATR-based design simplifies and modularizes the debugging of robotic systems. Hence, we decided to keep it for further analysis.

7. Ergonomic design (Antonelli et al. 2013; Morag 2003; Vozchikov 2013) deals with interactions between people and the product. As some interaction occurs between the team members and the robot in the testing and operating the robot, we decided to consider this method further.
8. Microprogramming (μP)-based design (Levin and Levit 1998; Baranov 1994; Agrawala and Rauscher 1976; Habib 1988; Shriver and Smith 1998) is used with products that include a microprocessor or a microcontroller. The μP-based design allows for designing the control by considering two different but dual (procedural and declarative) representations that ease designing, debugging, and coding, simultaneously (Baranov 1994; Levin and Mioduser 1996; Levin et al. 2001). The dual representations of control schemes allow using one that is more "natural" to describe the robot operation and the second that is better to address robustness and efficiency. The robot's control is based on a microcontroller, so we kept μP to the next stage.
9. Industrial design (Gemster and Leenders 2001) is concerned with the visual appearance of the product and the way it interfaces with the customer. These two are irrelevant to our robot and were not considered further.
10. Fault tolerance (FT) design (Abramovici et al. 1990; Mangir and Avizienis 1982; Avizienis 1976; Nelson 1990) is crucial for creating robust products and is inseparable method of obtaining good design. It provides insight for turning products that adhere to requirements and making them robust against some degree of faults. FT also demonstrates that in unstructured environments, no design could survive without making it robust to faults because it is usually impossible to foresee all potential situations. Clearly, we moved FT to the next stage.
11. Creative thinking methods (e.g., ASIT—Advanced Systematic Inventive Thinking; Horowitz 1999) support finding solutions to non-trivial problems that are manifested in all design stages. We kept it for further analysis.
12. Design for serviceability (Carter et al. 1964) is concerned with the ease with which maintenance can be performed on a product. Products often have parts that are subject to wear and are expected to be replaced at periodic intervals. The robots built by the students are not a product that is intended for an extended use. Hence, we disqualified this method for further analysis.
13. Fuzzy logic (FL) (Zadeh 1965; Lee 1990) helps in simplifying the control description. It is more straightforward and can be checked easily in comparison with other control methods. It is more intuitive to students and is faster to implement than other control methods. Fuzzy logic control design is used successfully in industry, and we thought it would be adequate to keep it for further consideration.
14. Design for the environment (Chen 2001) is concerned with issues such as recycling, environmentally friendly materials, product waste minimization, packaging recovery, and noise reduction. Some of the robot parts are reused from previous years' materials; nevertheless, this is not a major concern, so we eliminated this method from further consideration.
15. Detail design (Mukherjee et al. 2002) deals with the stage after conceptual design realizing the product. Obviously, tools related to this stage are important.
16. Design for manufacturability (Pitchumani 2005; Atkinson 1985) is not considered further as the robots will not be manufactured beside for the project.
17. Usability design (Iwarsson and Stahl 2003; Göransson et al. 2003) involves fitting the product to user's physical attributes and knowledge, simplifying user tasks, and making the user controls and their functions obvious. This is irrelevant for our purpose.
18. Design for reliability (Youn et al. 2003; Baily and Yin 2009) is quite similar to FT (item 10), which makes it redundant, consequently, not considered further.

After the initial assessment, 11 topics and methods remained as candidates: product architecture design, configuration design, parametric design, conceptual design, atomic requirements design, ergonomic design, microprogramming-based design, fault tolerance-based design, ASIT creative thinking-based design, fuzzy logic-based control design, and detailed design.

3.5 HOQ analysis

We used the HOQ, see (2) in Fig. 2, for selecting the design topics and methods according to the criteria presented in Table 1. The criteria were treated as the requirements and the design topics and methods as the engineering characteristics.

Table 2 presents the HOQ for choosing the appropriate design topics and methods. While this is not the classic use of the HOQ, it could be done without reservations. Based on the

Table 1 Robot's performance evaluation criteria

	Criteria		Criteria
1	Success in the contest	8	Fast navigation to all rooms
2	Driving well in corridor	9	Overcoming uneven floor
3	Making 90 and 180 degrees turns	10	Obstacle avoidance
4	Driving well in reverse mode	11	Non-tethered robot operation
5	Finding a white line on a black background	12	Sound activation of the robot
6	Finding a lit candle in a room	13	Navigation from each room back to starting point
7	Fast extinguishing of a lit candle		

criteria, the "whats" are listed in room 1. Room 4 lists the various design topics and methods that should be checked against the criteria. Next, we turn to room 2. The *criteria importance* was established by interviewing teachers and mentors and allocating the views along a 1–5 scale, where 5 is the highest. The previous years' robots were ranked according to the way in which they satisfied the requirements, on a 1–5 scale, and subsequently, the planned robots were rated against the requirements. The ratio of the planned to previous robots is called the *improvement ratio*. The product of *criteria importance* and *improvement ratio* gives the *total improvement ratio*. The *relative weight* is a normalized value of the *total improvement ratio*. The relationship matrix, room 4, shows how each design method helps attain the criteria list. A strong impact is worth 9, a medium high impact 5, a medium low impact 3, and a weak impact 1. The importance of the design topics and methods in room 5 is determined by multiplying each of the cells in the matrix by its *relative weight* and summing each column to give the *absolute importance*. The *relative importance* is the normalized *absolute importance*. Six methods rank highest and almost twice as high as the next in line: CD, FT, ATR, ASIT, FL, and µP. We selected them as the final six topics and methods.

3.6 Failure analysis and main problems encountered with previous robots

Another method used for selecting the design methods was failure analysis, see (3) in Fig. 2. We reviewed many of the

Table 2 HOQ of design topics and methods

	Room 4 → / Room 1	Room 3: Conceptual design	Ergonomic design	Product architecture design	Atomic requirements design	Microprogramming design	Fault tolerance design	Parametric design	Configuration design	ASIT creative thinking method	Fuzzy logic for robot control	Detail design	Room 2: Criteria importance	Previous years robots	Planned robot	Improvement ratio over previous robots	Total Improvement ratio	Relative weight
1	Performance	9	1	9	9	9	9	9	9	9	9	9	5	4	5	1.3	6.5	0.058
2	Realtime hardware failure resistance	9	1	1	5	5	9	1	3	5	9	1	5	2	5	2.5	12.5	0.111
3	System simplicity	9	1	3	9	9	5	1	1	9	9	1	4	3	4	1.3	5.2	0.046
4	Flexibility	9	1	9	5	9	9	5	1	9	9	1	4	2	5	2.5	10.0	0.089
5	Robot reliability	9	1	3	9	5	9	5	5	5	9	3	5	3	5	1.7	8.5	0.076
6	Software modularity	5	1	1	9	9	9	5	1	5	3	1	3	2	5	2.5	7.5	0.067
7	Robot testing ability	9	1	9	9	9	9	1	1	9	9	3	4	2	5	2.5	10.0	0.089
8	Fast hardware fixing	9	3	9	9	1	5	1	3	9	1	1	4	2	4	2.0	8.0	0.072
9	Ability of upgrading	9	3	3	1	9	9	9	1	9	9	9	2	2	5	2.5	5.0	0.045
10	Cost saving	9	1	1	9	9	9	1	1	9	9	3	3	1	5	5.0	15.0	0.134
11	Ease of transferring the subject matter	9	1	1	9	5	5	1	3	9	9	1	5	3	5	1.7	8.5	0.076
12	Short learning time	5	3	3	9	5	5	1	1	9	9	1	5	5	5	1.0	5.0	0.045
13	Ease of use	9	1	5	9	9	9	1	1	9	9	3	5	5	5	1.0	5.0	0.045
14	Can be modified to high school students	9	1	3	9	9	9	3	3	9	9	3	5	5	5	1.0	5.0	0.045
	Absolute importance	8.53	1.32	4.16	7.82	7.17	8.03	2.84	2.37	7.97	8.00	2.6	60.8				111.7	0.998
	Relative importance	**0.14**	0.02	0.07	**0.13**	**0.12**	**0.13**	0.05	0.04	**0.13**	**0.13**	0.04	Room 5					

previous robots' documents and reports, including interviews with teams, recalled failures of robots from previous competitions, and successes in local and international competitions. Upon organizing and sorting the data, we found the following as the main problematic issues.

1. Need for several hardware and software changes and modifications. We found that it was common among many teams to totally redesign their robot more than once. The most appropriate solution to this kind of problem would be implementing conceptual design methods and configuration design.
2. Malfunction equipment. Sometimes robots are not qualified in their trial runs due to malfunctioning equipment. A solution to this could be to introduce checkers that identify sensor failure and update the robot control.
3. We observed that high school students in general had difficulties in designing reliable robot speed and position control. The students had difficulties to calculate or experimentally find the proper gains of the PID control loop and were not aware what was happening with the robot control. They knew control theory, but they knew neither the essence of it nor how to decide on proper gains. In some cases, the improper gain values caused the robot to be too slow or too fast, and consequently, the robot hit the wall. The use of fuzzy logic control could remedy these difficulties.
4. When students reached the design stage, they stated the robot requirements among their team members in ambiguous ways. There was also inability to test and debug the robot because of contradicting or unclear requirement definitions. The ATR method would address these problems.
5. Occasionally, the teams did not overcome encountered problems properly. Solving these problems was possible using creative thinking methods such as ASIT.
6. The last noticeable group of problems was the difficulty to design and debug the robot control in a way that covers all possible situations. μP allows integrating a number of control representations and associated algorithms that remedy this situation.

The above analysis strengthens the previous selection of the six design methods.

3.7 Failure mode and effect analysis of adapting design methods for high school students

In order to reduce the chances of failing with these methods in the high school context, we exercised FMEA trying to think of the issues that could fail the methods and generate countermeasures, see (4) in Fig. 2. As the students were inexperienced, we had to adapt CORDEP to their level of engineering mathematics skills and experience. Another critical issue was the modification of industry development methods to suit the teaching environment of a high school where students lack prerequisite knowledge. Next, we describe the modifications made to each of the design topics or methods for their inclusion in the course material.

Conceptual design The teaching of CD requires no prerequisite knowledge; however, the time constraint forced a short version to suit the needs of the students. The stages of problem definition, and identifying customer needs with subjects, such as how to interview customers, using focus groups, preparing customer surveys, and handling customer complaints, were not taught because contest rules can be regarded as stating the problem and covering the customer needs. Only a small part of benchmarking was taught, as there was no identical commercial product to test against. There were robots from the previous year, which were analyzed by the teams in comparison with their robots.

Creative/inventive thinking ASIT was taught completely as it requires no special background and could easily be taught to the students in a short time. Another assisting factor in using ASIT was that we had an accessible simple training material that could be distributed to students for home practice.

Fuzzy logic As the designers were high school students, no deep mathematics background was introduced. The FL control subject was introduced as a technical straightforward procedure. The students learned to create the different membership functions, adapted to the capabilities of the microcontroller they used; derive the fuzzy rules; and receive the output variable for further processing.

Robot control Robot control was taught using an innovative teaching method built upon the use of dual representations (Levin et al. 2004a). The method was taught without the intensive mathematical manipulations. It is further explained through the microprogramming subject.

Atomic requirements This method was taught completely; it requires no special background and could easily be taught to students in a short time.

Microprogramming μP is an approach to teaching a number of subjects related to computer hardware. We adapted μP for designing robotic systems. The main idea of this adaptation is based on considering a robotic system to be a composition of two units: *a control unit* and an *operational unit* (Baranov 1994). The operational unit of the system includes such building blocks as motors, sensors, lamps, and manipulators. The control unit receives information from the operational unit and produces a sequence of control signals that results in executing desired operations by the operational unit. Usually, μP is a subject that is studied at the undergraduate level. It is built on a number of strong prerequisites including introductory logic design and programming. For introducing the subject into the high school robotics course, we developed a specific μP curriculum including a number of formal notations and definitions. The curriculum skips some technical details connected to specific computer architectures. Further, the presented μP concept includes only a finite state machine (FSM)-based microprogrammed controller and not the classical Wilks architecture. It allows presenting the concept of μP in a simpler manner and makes it practically productive for the process of robotics design.

Fault-Tolerant Design FT deals with a number of issues: verification, design for testability, built-in-self-test, and concurrent checking and more. These are useful in robotic systems design. Particularly, the robotics design described in this study includes one significant component for FT, which is the self-checking design (Levin and Karpovsky 1998). Within high school curriculum, the self-checking design was based on the development of specific redundant units, so-called checkers. The main goal of the checker is to prevent entering incorrect data to the control and operation parts of the robotic system. Students are able to construct checkers for robotic systems by using a number of standard solutions for the checkers design. These solutions are based on fundamental principles of so-called totally self-checking design: fault secure property and self-testing property. Students had to develop an appropriate checker and also prove its totally self-checking.

Within the current course design, the mathematics involved with FL and μP was too complicated. Yet, even by eliminating the mathematical details, there was sufficient benefit to teach these methods and use them. We considered teaching neural networks control but found it too complicated and of little importance. We also considered teaching 3D modeling and schematic software, but the teaching overhead and the software cost would not justify their inclusion. Finally, we considered teaching optimal product concept generation (SOS; Ziv-Av and Reich 2005) but found it too complex to fit into the course curriculum.

3.8 Chosen methods for the development process

To conclude, besides the general confidence about introducing design methods into the classroom, we used three guidelines to design the development process to teach: (1) addressing poor design practice by previous years' teams; (2) introducing methods that had high impact on attaining course's goals; and (3) avoiding complex methods. The six design methods selected are complementary and cover the complete development process; they include CD, ASIT, ATR, FT, μP, and FL.

Within the scope of the possible robotics design methods, these have an important role or influence over the product quality and its performance in real-world conditions. Moreover, these methods allow appreciating issues beyond the original goals. For example, FL allows appreciating that mathematics is not always about precise numbers. In fact, a great deal of engineering reasoning is qualitative and imprecise (Subrahmanian et al. 1993). Fuzzy control demonstrates that imprecise concepts lead to very robust behavior that is relatively easy to attain.

Subsequent to identifying the design methods, two experts used AHP to prioritize the methods in order to allocate them the necessary teaching resources (see (5) in Fig. 2). It was agreed that CD is the most important method (importance 42 % out of 100 % for one expert and 34 % for the second). The method that was secondly important was FT (19 and 22 %, respectively). The expert agreed on the following four methods but differed in the order of importance that they assigned to each method. Nevertheless, the expert assessment and our own judgment were quite consistent. After the relative importance evaluation, and given the stringent teaching hours limit, we decided to teach subsets of these design methods that deemed critical to the robot design or that would contribute significantly to other course goals. The findings and the experience from the first year of conducting this research prove the effectiveness of these methods.

While we selected six design methods for the robotics development process, we do not describe a particular sequence in which the tools should be used. This is because if each is clearly described with its input and output, then designers could use them effectively when the situation that calls for a particular tool arises. A particular design process only emerges out of a particular problem. In the Sect. 5, Figs. 3 and 4 illustrate the use of the methods in one particular robotic project.

4 Robot development process assessment

The effectiveness of the development process was assessed in multiple ways, both quantitatively and qualitatively. The assessment process demonstrates that it is possible to exercise extensive evaluation even in design studies—a contribution toward design science.

First, the robots developed were assessed through their performance in the high school division in the international Trinity College Fire Fighting Home Robot Contest (Ahlgren and Verner 2002; Ahlgren 2001) that took place on April 17–18, 2004, in Trinity College, Hartford, Connecticut.

The second evaluation was the researchers' observations of robot achievements in the contest, which included observation of the number of times and the time in the day that the qualification was completed, and comparisons between the special bonuses obtained by the robots in each run.

The third part of the evaluation included experts' questionnaire of the robot performances. The forth part

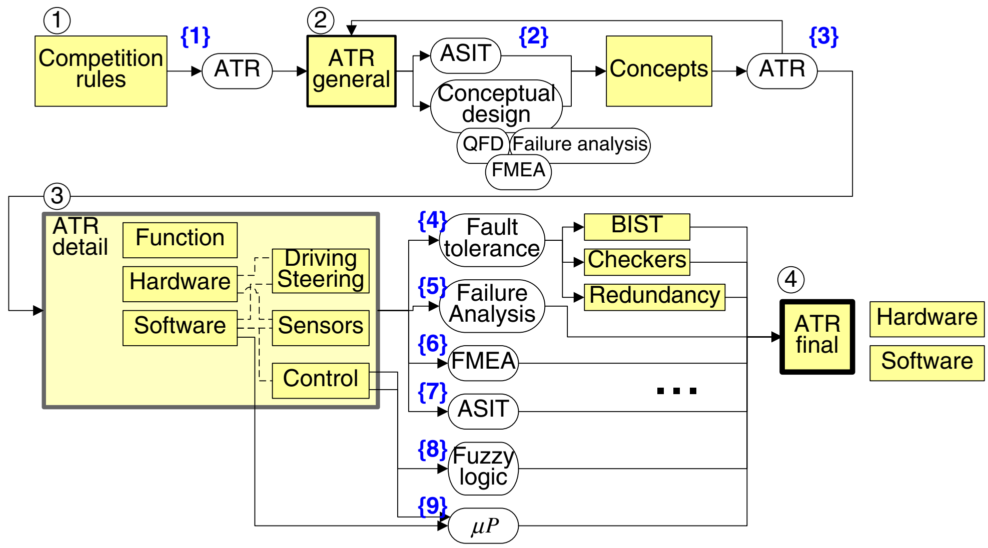


Fig. 3 Overview of the design process

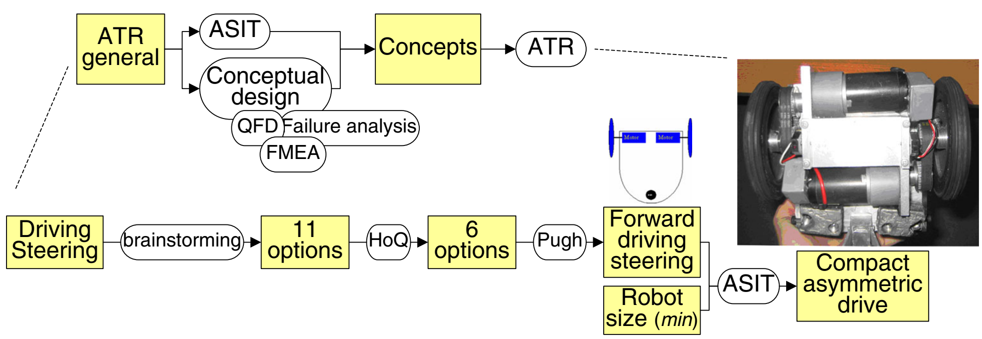


Fig. 4 Details of conceptual design of drive mechanism

included the experts' questionnaire of the students' design skills quality. The fifth part included experts' interviews about the students' design skills quality, and the sixth part included students', teachers', and experts' interviews about the design methods contribution to the project and to the students.

Most of the research was based on quantitative data with the addition of some qualitative insights we gathered from interviews with the students, the teachers, and the experts. Some of the data were analyzed using statistical tools.

4.1 Population

The research population included 127 high school students from four different comprehensive high schools. All the students were science majors and participated in a robotics course. The student profiles were similar in the population, and the students completed a mobile robot project. The schools are located within the center of the country and will be named 1, 2, 3, and 4. Some of the teams (73 students) designed firefighting robot (FFR) for the international robotics contest, and some (54 students) designed other mobile robot projects. Table 3 presents the number of students for each project type in each school.

Table 3 Number of students participating in robotics projects divided by school

School	Students participating in FFR project	No. of FFR teams	Students participating in other robot projects	No. of teams doing other robot projects
1	25	2	29	4
2	14	2	–	–
3	10	1	25	4
4	24	2	–	–

4.2 Teaching procedure

Four experts in the field of robotics and design were given the six methods and asked to prioritize the methods according to their importance for the requirements. The experts used AHP and gave their opinions about the importance of each method (Kolberg et al. 2007a). The first

Table 4 Research settings

School	Conceptual design	Fault-tolerant design	Fuzzy logic	Atomic requirements	Microprogramming	ASIT creative thinking method
1	✔	✔	✔	✔	✔	✔
2	✔	✔	✔	±		
3	✔					
4						

✔ subject learned, ± subject not learned but common technical documentation learned instead

expert was a PhD student in the mechanical engineering department of Tel Aviv University and had 15 years of experience in the robotics industry. The second expert was a CTO in a robotics company for 23 years and has an MSc in electrical engineering. The third expert worked for 29 years in a military unit that deals with autonomous military robots and holds a PhD degree in mechanical engineering. The fourth expert was a chief engineer with over 20 years of experience in a robotics company that developed robots for domestic use. We used their opinion to choose which methods to teach in schools 2 and 3, i.e., to choose the specific setup among many other possibilities.

The schools were randomly chosen for the teaching setup, which was implemented as follows: In school 1, the new design process was fully taught to all the robotics teams. In school 2, only part of the six methods was taught: μP and ASIT were excluded. ATR was also omitted, but regular documentation requirements were taught instead. In school 3, from the six methods, only CD was taught, and in school 4, the design methodology was not taught at all. In the schools where the design process was taught partially or not at all, the students learned the traditional design method as presented by the Israel Ministry of Education. Altogether, each student received the same amount of training in design methods as his peers.

Table 4 summarizes the research teaching setup. While we could have compared only school 1 and school 4, the additional two schools bring additional insight about the methods selected, by allowing a graded transition between the proposed development process and the traditional process. The students did not know about the differences among the schools or any other detail related to the research. From their viewpoint, they learned design as part of their robotics course, which prepared them to build robots for the contest.

4.3 Tools

4.3.1 The teaching methods

All the students were taught using a number of methods: frontal teaching, laboratory experiments, peer teaching in teams, self-teaching, project-based learning (with guidance), and contest-oriented learning (directed teaching). In frontal teaching, the students learned the traditional curriculum and the new development process methods. The teaching methods and the learning experience are described in Reich et al. (2005).

Table 5 Data collection research tools

Dependent variables	Data collection tools
Robot achievements in the firefighting international robotics contest	Trinity College official contest results
	Researcher's observations
Robot performances' quality	Experts' questionnaire
	Experts' observations
Students' design skill quality	Experts' questionnaire
	Experts' observations
	Experts' interviews
The design methods contribution to the project and to the students	Students' interviews
	Students' project reports
	Researcher observations in case study
Students' success in science subjects	Junior year students' grades
	Matriculation students' grades
Students' attitudes toward technology	PATT questionnaire pre- and post-scores
	Students' interviews

4.3.2 Data collection

Table 5 presents the dependent variables and the related data collection tools. There was no difference between data collection tools regarding the dependent variables.

In order to obtain a representative score for robot performances, the expert evaluations were coded with a questionnaire that included scores for each criterion (Table 1), from each expert, for each of the four schools. The scores were then analyzed to determine absolute and relative performance. The experts had no previous knowledge of the schools and the students. They had no idea about the new design methodology. They were not connected to the research in any way other than for evaluating the robot performances and designs. The experts heard about the contest but never came to see one before. They knew that the participants are high school students that learned a robotics course and built robots for the contest. The experts were asked to evaluate the robots that were participating in the contest. In order to analyze the students' design skills, data were collected from the experts' examination of the robot, experts' questionnaire of the students' design skills in relation to specific design criteria, and additional insight from the experts' interviews of the students' design skills.

The experts examined the robots at the local schools, testing them and watching films of the robots in action. The robot designs were evaluated according to the criteria listed in Table 6. Here, the robot itself was examined and not its performance, which was evaluated with tools described above. The criteria for evaluating the robots' design were selected based on the following considerations:

1. Experience gathered by the authors during several years indicated that these criteria are quite significant for project success.
2. The experts fully agreed on the criteria listed in Table 6.
3. Most of these criteria appear in the literature related to robotics.

At the subjective level, interviews conducted with the students, teachers, and experts throughout the project, along with detailed examination of the projects including the students' reports, allowed for evaluating the design methods. Data included students' opinion on the process and on their design skills improvement. Each project aspect was examined. The contribution of each design method was recorded.

Table 6 Students' design skills evaluation criteria

	Criteria
1.	Hardware failure resistance in real time
2.	System simplicity—hardware and software
3.	Fast changes (flexibility) possibility, due to concept or design changes
4.	Robot reliability in unpredicted environment conditions
5.	Software modularity
6.	Robot self-testing ability
7.	Students' off-line testing ability
8.	Fast hardware fixing ability
9.	Ability of upgrading to new robot and environment equipment
10.	Contribution to cost saving
11.	Overall score of robot design
12.	Hardware evaluation
13.	Software and algorithm evaluation
14.	Robot design average evaluation

4.3.3 Data analysis

Table 7 presents the data collection tools and the related analysis tools/methods.

4.3.4 Students' grades in science topics

We were interested in the impact of learning design methods and their practice on improving students' knowledge in science topics as well. In order to analyze that, we used students' matriculation grades as post-learning grades and the grades at the end of junior year as the pre-learning grades. *t* tests of the post–pre-grade differences in each science topic split by schools and a generalized linear model (GLM) test were used to find whether there were differences between the schools in the junior year pre-grades.

4.3.5 Students' attitudes toward engineering/technology

We were also interested to know whether the suggested design methodology had any influence on the students' attitudes toward engineering/technology. For that, we used the Pupils Attitude Towards Technology (PATT) questionnaire, which was tested and validated in at least 12 countries as appears in Boser et al. 1998; Becker and Maunsaiyat 2002; Volk et al. 2003 and also in Israel (Betzer 2002).

We made *t* tests of the questionnaire filled out by the students at the end of the project compared to the questionnaire they filled at the end of their junior year, split by schools. We also used GLM test to find whether there were differences between the schools in the junior year questionnaire data.

4.4 Implementation

The teaching of the full or partial methodology in three schools and traditional methods in school 4 was conducted in parallel. The parallel tracks ended with measurement, evaluation, and comparison between the four schools.

In 2003, a pilot study with two teams from the same school was carried out. The study led us to some changes in order to further adapt the methods to high school students and environment that is discussed later. In 2004, we taught the robotics course in all schools and conducted the full-scale research.

Table 7 Data collection analysis tools/methods

	Data collection tool	Analysis tool/method
1.	Trinity College official contest results	Comparison of robots' ranking with other robots and teams
2.		Comparison of absolute robots' scores with other robots and other contest divisions
3.	Researchers' observations of robot achievements in the contest	Observation of the number of times and the time in the day that the qualification was completed
4.		Comparison between the special bonuses obtained by the robots in each run
5.	Experts' questionnaire of the robot performance	Comparison of the robot performance scores between schools
6.		GLM (general linear model)–analysis of variance (*F* test) and comparison between schools
7.	Experts' questionnaire of the students' design skills quality	Comparison of the students' design scores between schools
8.		GLM–analysis of variance (*F* test) and comparison between schools and comparison between the scores the experts gave to the various robot designs
9.	Experts' interviews about the students' design skills quality	Qualitative analysis of the expert insights of the student designs
10.	Students', teachers', and experts' interviews about the design methods' contribution to the project and to the students	Analysis of the students', teachers', and experts' insights on the contribution of the design methods to the students and to the robot performances and extraction of the most important factors in each design method
11.	Students' project reports	Analysis of the design method contributions and extraction of the most important factors
12.	Detailed case study	By showing a detailed case study, the methodology implementation in the field is discovered and we can see actual solutions to diversity of problems and many design issues' implementation and that add insight about the design methodology
13.	Junior year and matriculation student grades	Paired-samples *t* test
14.		GLM–analysis of variance (*F* test)
15.		Linear regression
		One-way ANOVA (analysis of variance)

Table 7 continued

	Data collection tool	Analysis tool/method
16.	PATT questionnaire pre- and post-scores	Cronbach's α reliability test
17.		Paired-samples *t* test
18.		GLM–analysis of variance (*F* test)
19.	Students' interviews	Analysis of the students' desire to continue learning or working in engineering areas

During this year, we conducted a research where the students worked on their robots in their senior high school year. The students had to fulfill all the regular requirements in all the topics of their studies. In 2005, the researchers were not involved at all. The teaching setup was different. School 4 students learned the full methodology, and school 1 students learned 3 out of the 6 design methods (as taught in the year of 2004 to school 2). The development process was taught by different teachers than in the previous year.

In 2005, the contest was changed significantly. One change was splitting the high school division into two divisions: entry level and standard level. One team from school 1 and one team from school 4 participated in the entry-level division, and another team from school 1 participated in the standard division.

5 Results

5.1 Case study examples

The complete robot design is too complex to review in a paper; therefore, we chose part of the design that includes contributions from all participating disciplines to illustrate the problems faced by the students and some of the design tools they used. The chosen part is the design of the driving/steering subsystem.

Figure 3 provides an overview of the design process. As was discussed before, this particular sequence of tool utilization is borne out of the particular design problem. Another problem might have resulted in a different application of methods. Initially, the team studied the competition rules and extracted requirements that were gradually refined with ATR (step {1} in Fig. 3). Each of the subjects that the students derived from the contest rules was further analyzed for obtaining more detailed "atomic" requirement. For example, the requirement for autonomous robot was further divided into subsystems. One such subsystem was the driving/steering mechanism. In relation to the driving/steering system, the students raised additional requirements that assisted them to appreciate the complete task. To illustrate, the students realized that a smaller robot could have more space to recover from undesired situations like hitting the wall. That directed them to design a small footprint robot and to increase its height for the needed hardware space, while maintaining reasonable robot stability.

After the requirement refinement, the students started using CD (step {2} in Fig. 3 that is further detailed in Fig. 4 for the driving–steering). They divided the requirements into four robot main subsystems (mechanics, electronics, software, and control) and proceeded with each subsystem, considering its own requirements, using tools such as HOQ, Pugh, ASIT, FMEA, and failure analysis. By the time the students finished with steps {2} and {3}, their initial subdivision into four main subsystems had changed to the one depicted in Fig. 3, as ATR detail ③.

The design of the driving/steering concept illustrates the students' conceptual design. They carefully considered 11 possible alternatives recording all pros and cons and the reasons for the final choice (described in details in the following section). Subsequent discussions focused on decreasing overall robot footprint size. We will demonstrate in details the steps that the students took in order to determine the best drive mechanism for the robot.

5.1.1 *Selection of concept*

The first step was looking at the contest rules and deriving initial requirements from these rules. Some of the contest requirements were as follows: autonomous robot, navigating through the arena as fast as possible, avoiding obstacles, negotiating the uneven floor item, and not touching the wall. Then, refinement of the requirements led, for example, to considering the driving/steering system design. This was further refined into more atomic requirements (see Table 8). The students then indicated the level of importance of each requirement. Every team member had to decide upon the rate from 1 (low importance) to 5 (high importance) of each requirement. Table 8 presents the results of the team.

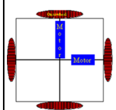
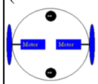
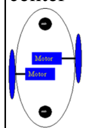
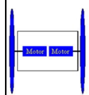
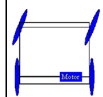
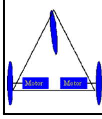
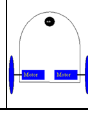
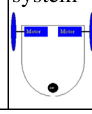
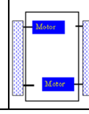
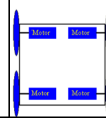
There are many types of driving/steering systems for robots (see Table 9). In many real projects, engineers quickly select one solution principle and continue the design without proper evaluation of other alternatives. In our case, the students built up a table with all the steering/driving systems they found in the literature and from inquiring experts. The students analyzed all the configurations with their pros and cons. For example, consider the arguments related to the tricycle driving. Pros: "Forward/backwards movement control is very easy. All it takes is directing the steering wheel to be parallel to driving wheels. When climbing over uneven floor item, the robot is stabilized because all three wheels will always touch the floor." Cons: "When making turns, each of the driving wheels makes a different path. That means that one driving wheel should rotate differently from the others. As they are connected to the same axle, this means that one wheel will be dragged. This will have an effect on the robot movement. The robot cannot make pivot rotations about its center of gravity. This makes it more difficult to navigate in a room with furniture."

Table 8 Requirements importance rating

	Requirements	Frequency of responses with 4 or 5 rating (%)
1	Maneuverability when making turns	83
2	Ability for fast recovery from the corridor middle straight line	81
3	No wall hitting	80
4	Ability to overcome inclined surfaces in a reliable way	75
5	Ability to avoid furniture with easy maneuver	80
6	As small as possible turn radius	77
7	As small as possible correction after doing a turn	84
8	Convenient reverse driving	60
9	Reliable approach to the candle within a room	85
10	Stable and will not crush if will hit a wall, furniture, or inclined surface	84
11	Fast and reliable aligning of the robot at all room entrances	86
12	Reliable aligning of the robot in front of the candle before extinguishing it	78
13	Robot stability during driving	86
14	Simple and easy to implement system	84
15	The ability of the software team to deal with the chosen driving/steering system	90
16	As small as possible footprint	82
17	Fast driving	78

The students chose to use the HOQ for selecting the best concept. Table 10 presents the HOQ made by the students. Based on the requirements shown in Table 8, the "Whats" are listed in room 1. Next, we turn to room 2. The team priorities were established by taking the results from Table 8 and converting them to a 1–5 scale, where 5 is the highest. In previous years' robots, the students studied these robots and ranked the level that the last years' robots driving/steering systems satisfied the requirements on a 1–5 scale. Then, the students rated the planned robot driving/steering system against the requirements and proceeded to calculate the relative weight of each requirement. Six requirements, shown in bold font in the last row, rank

Table 9 Driving mechanism types

XY 	Symmetrical center (stabilized) 	Asymmetrical center 	Symmetrical center (not stabilized) 	Ackerman drive (car) 	3X3 drive 
Tricycle 	Backward driving/steering system 	Forward driving/ steering system 	Caterpillar drive (track as in tank) 	4X4 drive 	

highest: 2, 9, 11, 13, 15, and 16. Room 4 lists the driving/steering systems that should be checked against the requirements. The students completed the house to obtain the relative importance in room 8.

The front driving/steering method obtained the best score. Notwithstanding, as some of the scores are close to the best one (marked with bold font), the students felt that the scores obtained from the HOQ are not decisive and need further consideration. Therefore, the students decided to follow Pugh's concept convergence in order to make their final concept decision. The results are presented in Table 11 showing the two best options, the front driving–steering and the 3 × 3 system. The students decided to choose the front driving–steering and not the 3 × 3 system because of the difficulty to build it and the complexity of the software needed for it.

5.1.2 Footprint reduction (use of ASIT)

The students realized that the smaller the robot footprint is, the more robust its navigation would be. When the robot is smaller, it will be easier to correct a deviation from the center of the corridor. Bigger robot might hit the wall before it will take action for recovering from the error. The students chose to use ASIT for minimizing the robot footprint. While doing this, the students took into account the driving–steering mechanism that was chosen and will be installed within the robot structure. The students considered several alternatives from which only the final solution is presented here. The ASIT process makes use of several tools that use predefined templates. The team considered the tools and decided, for example, that the "object removal" tool seemed inadequate; "breaking symmetry" might work; and "division" would be the second tool to try. The template for the "breaking symmetry" tool used by the students is introduced next.

I. ASIT preparation stage

Problem objects list: motors, wheels, connectors, base plate.
Neighborhood objects list: robot components, arena
Functional structure: The robot needs to have small footprint. The objects that restrict the robot size are motors. Front driving/steering system requires that two front wheels will be parallel and at the robot front. Aligning the two DC motors in front with the same geometrical axis will cause relatively large footprints.

II. ASIT solution stage

- *Operation*: decreasing robot footprint
- *Strategy selection*: restructuring
- *Restructuring technique selection*: breaking symmetry
- *Select an object*: motors
- *Form important list of object parameters*: longitudinal dimension, diameter, and material.
- *Solution statement*: The object **motors** will be modified so that the object's parameter **longitudinal size** will be related to it in the following way: decreasing longitudinal dimension.

The above process is quite straightforward, and the solution became obvious when going through all the steps. In order to implement the longitudinal dimension reduction, the team finally decided to design the motor block so that the motors would be parallel to each other, and the wheel axle will come from the center between these motors as appears in Fig. 5. The wheels–motor assembly is presented

Table 10 HOQ of driving–steering system

		XY	Symmetrical center stabilized	Asymmetrical center stabilized	Symmetrical center not stabilized	Ackerman	Tricycle	Rear driving/steering	Front driving/steering	Caterpillar	4 × 4	3 × 3	Team importance	Previous years' robots	Planned robot	Improvement ratio over previous robots	Total improvement ratio	Relative weight
1.	Maneuverability when making turns	1	9	9	1	3	3	9	9	9	9	9	4	4	5	1.3	5.2	0.06
2.	Ability for fast recovery from the corridor middle straight line	1	9	9	1	1	1	9	9	9	9	9	4	3	5	1.7	6.8	0.07
3.	No wall hitting	1	3	3	3	1	1	3	3	1	3	3	4	3	4	1.3	5.2	0.06
4.	Ability to overcome inclined surfaces in a reliable way	1	1	1	9	1	1	3	9	9	3	9	3	3	5	1.7	5.1	0.05
5.	Ability to avoid furniture with easy maneuver	3	3	1	1	9	9	3	9	3	3	9	4	3	4	1.3	5.2	0.06
6.	As small as possible turn radius	1	9	9	3	1	1	9	9	9	9	9	3	3	4	1.3	3.9	0.04
7.	As small as possible correction after doing a turn	1	9	3	1	1	1	9	9	9	9	9	4	4	5	1.3	5.2	0.06
8.	Convenient reverse driving	9	9	9	3	3	3	9	9	9	9	3	1	4	4	1.0	1	0.01
9.	Reliable approach to the candle within a room	1	3	3	1	3	3	3	9	3	3	9	4	3	5	1.7	6.8	0.07
10.	Stable and will not crush if will hit a wall, furniture, or inclined surface	9	3	3	1	3	3	3	3	9	3	3	4	4	5	1.3	5.2	0.06
11.	Fast and reliable aligning of the robot at all room entrances	1	9	3	1	1	1	9	9	9	9	9	5	3	5	1.7	8.5	0.09
12.	Reliable aligning of the robot in front of the candle before extinguishing it	1	9	3	3	3	3	9	9	9	3	9	3	3	4	1.3	3.9	0.04
13.	Robot stability during driving	3	3	3	1	9	9	9	9	9	3	9	5	4	5	1.3	6.5	0.07

Table 10 continued

	XY	Symmetrical center stabilized	Asymmetrical center stabilized	Symmetrical center not stabilized	Ackerman	Tricycle	Rear driving/steering	Front driving/steering	Caterpillar	4 × 4	3 × 3	Team importance	Previous years' robots	Planned robot	Improvement ratio over previous robots	Total improvement ratio	Relative weight
14. Simple and easy to implement system	1	9	9	1	1	1	9	9	1	1	1	4	3	4	1.3	5.2	0.06
15. The ability of the software team to deal properly with the chosen driving/steering system	1	9	3	1	1	1	9	9	3	1	1	5	4	5	1.3	6.5	0.07
16. As small as possible footprint	1	1	9	3	1	1	9	9	3	1	1	4	2	5	2.5	10.	0.11
17. Fast driving	3	9	9	1	3	3	9	9	3	9	9	3	4	5	1.3	3.9	0.04
Absolute importance	1.92	5.98	5.18	1.94	2.62	2.62	7.38	8.46	6.12	4.8	6.48	53.5				94.1	1.02
Relative importance	0.04	**0.11**	**0.10**	0.04	0.05	0.05	**0.14**	**0.16**	**0.11**	0.09	**0.12**						

in Fig. 6. The students continued to use CORDEP and completed the first robot prototype shown in Fig. 7.

Another issue that had to be addressed in the initial design was the wheel diameter and width. First, after watching the videos of the previous year, they measured the speed of the winning robot and it was around 0.4 m/s. They decided upon 0.45 m/s as the regular speed. The motors' speed was fixed to 100 rpm, which gave the control mechanism some clearance for correction to the up and down directions. Therefore, the wheel diameter was determined by Eq. (1).

$$D[\mathrm{cm}] = \frac{V[\mathrm{m/s}] \times 60[\mathrm{s/min}] \times 100[\mathrm{cm/m}]}{S[\mathrm{rpm}] \times \pi}$$

$$D[\mathrm{cm}] = \frac{0.45[\mathrm{m/\sec}] \times 60[\mathrm{s/min}] \times 100[\mathrm{cm/m}]}{100[\mathrm{rpm}] \times \pi} = 8.6[\mathrm{cm}] \quad (1)$$

The wheel diameter was actually closer to 9 cm, so minor changes in the control software were needed.

5.1.3 Change management

In reality, design changes arise throughout the life cycle of products. It is interesting to mention one instance in which the team addressed such a change in a situation where other teams from around the world seemed helpless (Kolberg et al. 2007a, b). When the team from school 1 first came to Trinity College on Saturday, the earliest time possible for qualification purpose, they noticed that the uneven floor items were different from those published in the official contest Web site, and the robot, which was designed for different items, would not perform well with these modified items. Figure 8 shows the floor item building instructions as were introduced by the contest official Web site. Figure 9 presents the actual floor item which was the published one (white material) covered with black plates that made the floor item higher and wider.

The students initiated an immediate ASIT session to find a solution. Again, one tool had to be selected and its template used. As there was a need to increase the robot stability, the time was short, so making major changes to the robot's construction was not a desired option. The caster wheel seemed to be the least problematic. It was also obvious that in order to increase the robot's stability and therefore increase the robot's size, an extension method will be chosen. As the students did not want to add materials at that point, they decided to try the unification technique whose template is summarized below.

I. ASIT preparation stage

Problem objects list: motors, wheels, connectors, and base plate.

Table 11 Driving–steering system selection using Pugh method

	Symmetrical center stabilized	Asymmetrical center stabilized	Rear driving–steering	Front driving–steering	3 × 3	Caterpillar
Maneuverability when making turns	S	S	−	S	+	DATUM
Ability for fast recovery from the corridor middle straight line	S	−	S	+	+	
No wall hitting	S	S	S	S	S	
Ability to overcome inclined surfaces in a reliable way	−	−	−	+	−	
Ability to avoid furniture with easy maneuver	S	S	−	+	+	
As small as possible turn radius	S	S	−	+	S	
As small as possible correction after doing a turn	S	S	+	+	+	
Convenient reverse driving	S	S	S	S	+	
Reliable approach to the candle within a room	S	S	−	+	+	
Stable and will not crush if will hit a wall, furniture, or inclined surface	S	S	S	+	S	
Fast and reliable aligning of the robot at all room entrances	S	S	−	+	+	
Reliable aligning of the robot in front of the candle before extinguishing it	S	S	S	S	S	
Robot stability during driving	−	−	S	+	+	
Simple and easy to implement system	S	S	+	+	−	
The ability of the software team to deal properly with the chosen driving/steering system	S	S	−	−	−	
As small as possible footprint	+	+	+	+	+	
Fast driving	+	+	+	+	+	
$\sum +$	2	2	4	12	10	
$\sum -$	2	3	7	1	3	
$\sum S$	13	12	6	4	4	

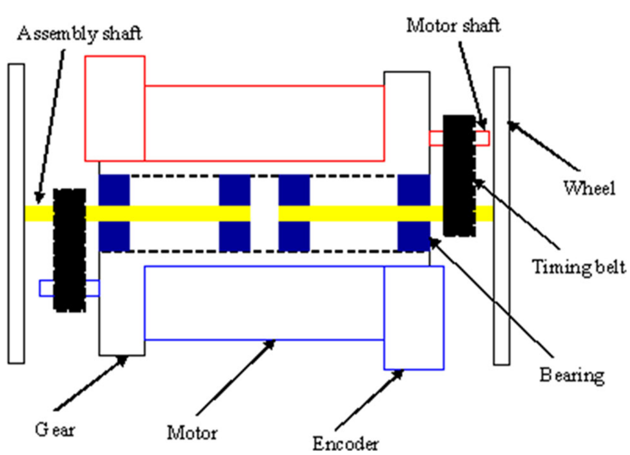


Fig. 5 Wheels–motor assembly designed by the students

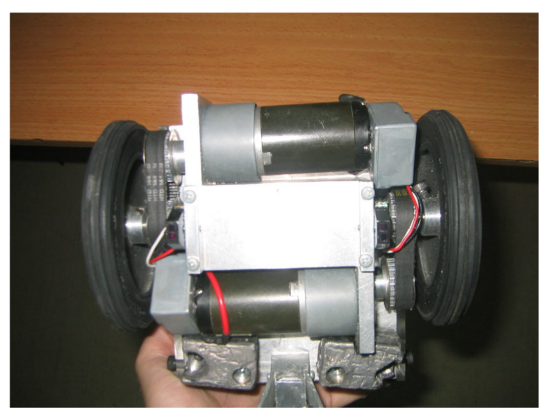


Fig. 6 Actual wheels–motor assembly

Neighborhood objects list: robot components, arena, and floor items
Functional structure: The robot has to overcome the new floor items. The object that decreases the robot ability to handle the new floor items is robot size.

II. ASIT solution stage

Operation: Increasing robot size
Strategy selection: extension

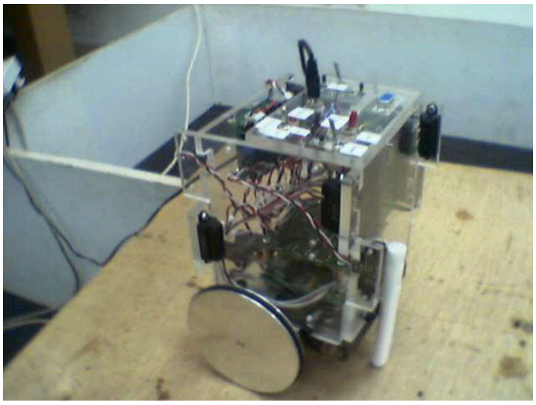


Fig. 7 First robot prototype

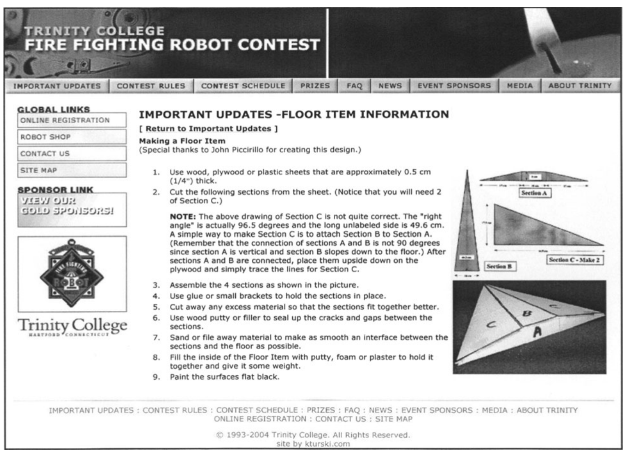


Fig. 8 Official floor item building instructions

Conceptual solution: The relation "decrease the robot ability to handle the new floor items" will change from decrease relation to increase if the following operation of "increasing the robot size" will be performed.
Restructuring technique selection: unification
Select an object: caster wheel
Solution statement: The object "caster wheel" will be modified so that it will increase the robot size.

The solution that evolved from this session was to move the balancing wheel in the radial axis toward the outer dimension by 2 cm. Realizing the risk of such change, we have to remember that it alters many of the navigating values, including turns, rotations, wall distance, and more. Nevertheless, the team was confident. The software team said it would not be a difficult problem, as they have built the software according to μP and FT guidelines. Therefore, it will take no more than 2 h including experiments to modify the software. This was in fact true; it took 20 min for the hardware team, and then, after 1 h and 30 min, the robot was fully functional and handled perfectly uneven floor items. This is not a trivial matter. The solution was found and implemented smoothly due to the following key points:

1. The student skills of organizing the information and analyzing the advantages and disadvantages of the alternatives as they learned in the CD enabled reuse of previous considerations.
2. The contribution of the ATR to handle both hardware and software components easily; for example, instead of the microinstruction Y9 (composed of the microoperations: y1,y2,y3,y4,y9,y12)—rotate robot in medium right arc turn, the students used microinstruction Y7 (composed of the microoperations: y1,y2,y3,y4,y9,y14)—rotate robot in small right arc turn (where y1 is turn on right motor, y2 is turn on left motor, y3 is rotate forward right motor, y4 is rotate forward left motor, y9 is rotate right motor at low speed, y12 is rotate left motor at medium (nominal) speed, and y14 is rotate left motor at high speed).
3. The clear division between hardware and software tasks, allowed the hardware team to work in parallel with the software team, which, in turn, decreased substantially the time needed for fixing the problem.
4. μP which allowed for easy debugging.
5. FT that helped students implement the desired change. The students made a detailed design regarding the robot balance beforehand. The change sent the students back to their calculations. They found that this change will improve the robot balance and will allow for more tolerance. On the other hand, they observed the increasing area of the base that forced them to feed new parameters to the software and check the robot ability to overcome faults. They found that FT principles such as escaping from a too close wall or navigation inside a room were still valid.
6. The efficient ASIT session the students made in order to solve the problem.

5.2 Evaluation of robot performances in the contest

The effectiveness of CORDEP was assessed in multiple ways. First, the best products of the design process competed in the Trinity College Fire Fighting Home Robot Contest (Ahlgren and Verner 2002; Ahlgren 2001) that took place on April 17–18, 2004, in Trinity College, Hartford, Connecticut. The teams had to overcome a qualification stage on Saturday, between 10:00 and 21:00 h, before the contest itself on Sunday. Each team had three trial runs. The robots needed to succeed in one of those runs in extinguishing a lit candle. In the qualification stage, there are only simple runs without any operating modes (like having inclined surfaces along the course or

Fig. 9 Actual floor item

obstacles within the "rooms"), as opposed to the contest itself during the next day.

The same contest also had a senior division where the participants were engineers, faculty, and engineering students and was open to every team that wanted to compete. The senior contest was identical to the high school contest, with the same rules, arenas, judges, and conditions. The organizers made this artificial separation because of the superior knowledge, skills, resources, etc., that engineers, students, and faculty have and high school students lack.

In the qualification day, school 1 qualified early in its first trial, school 2 qualified in its second trial, school 3 qualified in its third trial, and school 4 qualified in its third trial in the last minute. The official results of the contest in 2004 are presented in Table 12.

The scores of the three top senior robots were 18.09, 34.01, and 125.67. We may see that the score of the first place of the high school league robot $ff is much better than the score of the senior league first place robot. Furthermore, if the robot Jimmy from school 2 would have participated in the senior league, it would have taken the third place.

The second team from school 1 did not participate in the contest due to misunderstanding in the registration, although they travelled from Israel to participate. Nevertheless, their robot was tested under the contest conditions in the same arena and received a score that would have given it second place in the contest. The other two FFR's teams did not participate in the international contest due to lack of funding.

In 2005, the complete process was taught to school 4 students. The team from school 4 that participated in the entry-level division won the first place. The team from school 1 (that learned part of the methodology) that participated in the entry division won the third place. The team from school 1 that participated in the standard division won the second place.

5.3 Experts' evaluation of the robots' performance

Four experts ranked the robots' performances according to the criteria detailed in Table 1. The experts scored each criterion. The results are summarized in Fig. 10. The four bars denote the average score obtained by the experts for schools 1, 2, 3, and 4, on the particular criterion, starting from the left. The average scores across all criteria of the experts, on a scale of 100, were 98.9 for school 1, 86.0 for school 2, 75.6 for school 3, and 67.5 for school 4. Statistical analysis of the scores concluded that there were significant differences between the school scores in average and for all criteria except for 3 and 13, where noticeable differences in the expert scores would not lead to significant differences between the schools. Criterion 11 is irrelevant for this test because all the robots had untethered operation.

Table 12 Official contest score results

School (robot)	Rank	Score (s)	No of successful runs	Modes used successfully[a]
1 ($ff)	1st	8.54	3	All
2 (Jimmy)	6th	62.7	3	Sa, ut in 3 runs, rt in one run
3 (Kaktus)	14th	1,214	1	Sa, ut (14 s for one run)
4 (Villa)	16th	1,222	1	Sa, ut (22 s for one run)

[a] *Sa* sound activation, *ut* untethered, *rt* return trip

5.4 Experts' evaluation of the robots' design

Four experts evaluated the robots' design. They examined the robots, project reports, contest achievements, and preliminary designs. They ranked the robots' design according to the criteria in Table 6. The results are summarized in Fig. 11. In average, the expert scores out of 100 were as follows: 96.4 for school 1, 86.7 for school 2, 73.4 for school 3, and 65.0 for school 4. Statistical analysis of the scores discovered that there were significant differences between school scores in average and for all criteria except for 10. Regarding criterion 10, there were noticeable differences among the experts regarding contribution to cost saving.

The experts were interviewed about their opinions on the robots' designs. We present few of their evaluation statements.

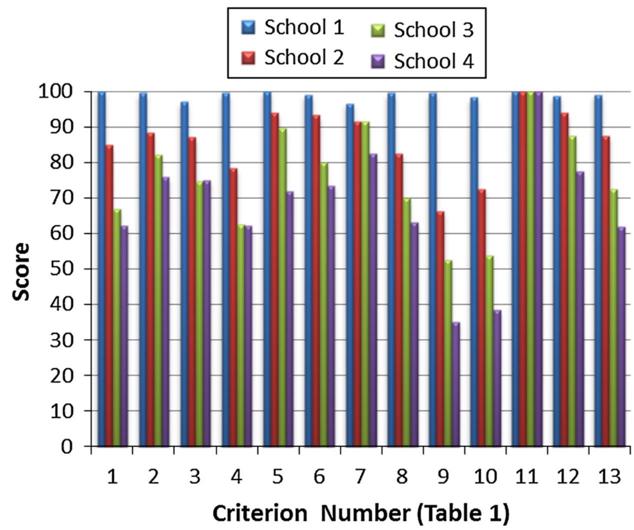


Fig. 10 Experts' evaluation of robot performances

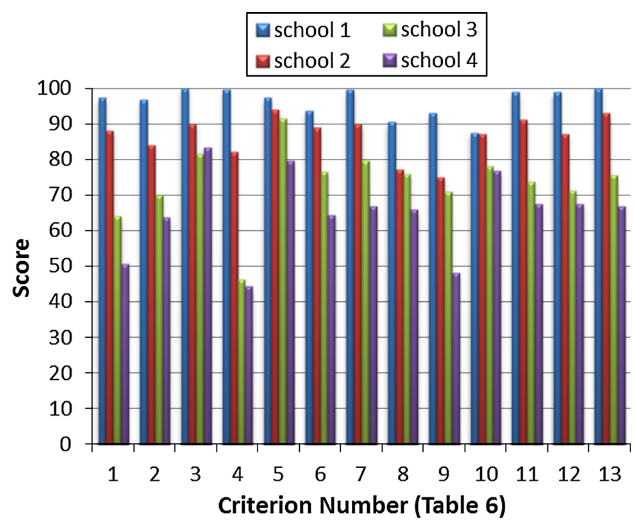


Fig. 11 Experts' evaluation of robot designs

Expert a: "The design of the robot $ff was remarkable. The students made every possible effort in a quite tight schedule to design and produce a robot that is robust, fully functional and flexible enough to be able to adapt to rule changes. The design of the robot Jimmy was very good. The hardware design was good, but there was a place for improvement. The design of the robot Kaktus was average. The hardware design was not fully robust, and some of its components were not well fastened. The design of the robot Villa was less than average and the robot was only partially functional."

Expert b: "I was impressed by the robot $ff. It is clear that a remarkable design work has been done. From my experience, it would fit more experienced engineers than high school students. Jimmy also looked very good. There were differences compared to $ff though. Its structure made it less convenient for fixing, replacing, or expanding with new hardware. Kaktus seemed to be less robust than the two I previously discussed. Villa was not well designed. It functioned partially and both its hardware and software need to be improved."

Expert c: "The robot $ff was outstanding. Its distance sensors locations were impressive. The robot Jimmy was well designed, with a unique triangle shape. The robot Kaktus was designed with less attention to details. The robot Villa has a poor design. Some of the sensors were missing for proper operation, wires hanged around all over the robot."

Expert d: "If I have to compare the robots, I would say that definitely $ff is the best in all design aspects. Jimmy is also a good one but lacks some robustness and ease of trouble shooting hardware. Kaktus has more flaws, concerning robustness, structured software code, and the ability to work on a hardware piece that is not in the external envelope. Villa has a problematic design."

5.5 Students' evaluating the contribution of design methods

Next, we present interviews conducted with the students about how they grasped the main contribution of these design methods to the success of their project execution. We describe these in details that non-experts in robotics might not fully understood; nevertheless, they demonstrate the profound implications that these methods had on the success of the robots' design and student capabilities.

1. Conceptual design
   a. Created good balance between quality requirements, cost, and time schedule up to the contest date;
   b. Supported multidisciplinary and interdisciplinary teamwork;
   c. Allowed for wise integration of creativity, analysis and synthesis, and determination of the chosen alternative out of discussion, consideration, persuasion, and agreement;
   d. Helped to find an excellent solution by weighting the design alternatives along many diverse criteria;
   e. Enabled flexible, open architecture, and tailored solution according to changing demands;
   f. Supported quantitative measurement of design quality; and
   g. Integrated the relevant techniques of the development process that support the whole conceptual design process.

2. Fault tolerance and design for testability
    a. Supported the understanding of logical fault models;
    b. Enabled fault detection and redundancy in diverse circuits;
    c. Enabled fault modeling as a design tool for fault detection and redundancy;
    d. Included testing and detecting single-stuck faults; that was made by a periodic check with the help of the interrupt handler routine and checking for fault sensors or fault motor controller software instructions;
    e. Supported functional testing for functional faults' detection; that was made also off-line, as a checking routine of all the robot subsystems, like motors' driving–steering operation;
    f. Directed toward a design for testability, determination of device state, rapid fault isolation, and development of tests that were cost-effective and reasonable for determining this state;
    g. Included built-in-self-tests (BIST; Abramovici et al. 1990)[5];
    h. Enabled self-checking design including checkers design; and
    i. Enabled detection of the location of logic level faults' location.
3. Fuzzy logic for robotic system control design
    a. Used less memory space, since the microprocessor includes a fuzzy kernel;
    b. Allowed for faster performance because it is more understandable to the students;
    c. Improved control reliability, because it would still work even with non-optimal term definitions;
    d. Worked under uncertainty conditions so it still functioned even with lack of information;
    e. Led to more intuitive PID (proportional, integral, derivative) control parameters' selection that required no expert knowledge of the system;
    f. Adapted easily to a new problem because the technique was simple to implement; and
    g. Required only basic mathematics knowledge, in order to implement the robot control.
4. Atomic requirements (ATR) design
    a. Allowed allocating an atomic requirement to exactly one system component, in a unique sense, thus reducing faults and error sources;
    b. Helped to create tests that would isolate problematic component causing fault;
    c. Yielded a possible direction for a solution in some cases when detailed elaboration of the system atomic requirements was done;
    d. Allowed for testing an alternative hardware or software, by functional ATR description, for easier and cheaper operation, and even components reduction;
    e. Enabled clear understanding of each component function and therefore easier system design and more understandable interface between hardware and software subteams;
    f. Fitted well for digital systems in the robot, such as the robot microcontroller and additional components; and
    g. Served as a preliminary stage for μP, such as when defining the functional inputs and outputs into and from the control unit.
5. Creative–inventive thinking throughout the design process
    a. Encouraged thinking and not just memorizing;
    b. Enabled teamwork versus individual work: All ASIT activities were done among team members; the ideas and tools were chosen by the group, where each was responsible for a different aspect of the robot within its subgroup of hardware or software;
    c. Supported multidisciplinary learning versus disciplinary knowledge acquisition (the problems the students dealt with were multidisciplinary; the solutions found with ASIT were also multidisciplinary including from remote domains);
    d. Led to generalization and integration of subject matters, as opposed to focusing on a specific area (as problems came from various disciplines and often were interrelated, the solutions had to be considered after an understanding of the system in general and specific topics integration as well); and
    e. Created the ability to distinguish between the essence and the subordinate (the items that were not relevant to the solution were removed in order to eliminate unnecessary information).
6. Microprogramming system design
    a. Brought an understanding of the microcontroller architecture and function (the separation of microcontroller structure into control unit and data path brought a new insight to the students and helped them understand the functions and operation of the microcontroller they used);
    b. Built a cognitive bridge between the hardware and software in microprocessor-embedded systems and enabled equivalence testing;

[5] BIST is an off-line test that is activated by a push button switch. When pushed, special test software is initiated, and when it finishes running, it reports the status of the robot's subsystems.

c. Allowed bridging between traditional logic systems approaches and processors μP-based approaches (the students observed the advantages of FSM in the software design and as a valuable debugging tool; when they built their algorithms by using ASM (algorithmic state machine), they collected all possible states and transitions between states; it is also useful for debugging, because each state could be checked separately; if a fault occurred, it was clear which state to check and fix the corresponding fault);
d. Suggested a synthesis of the hardware and software theories; the hardware was presented by a microcontroller as a control unit and data path; the software was presented using ASM and FSM along with the interrupt mechanism. These presentations created a complete view of the hardware, software, and the relations between them;
e. Detailed a clear description of digital systems with functional separation including (1) control unit, which was composed of standard logic elements, (2) the data path that contained various sorts of elements, and (3) the memory function that is related to both; and
f. Enabled reduction in hardware components or software code: The students learned how to combine two ASMs or FSMs into a smaller combined one.

5.6 Students' achievements in science topics

Paired-samples *t* test (split by schools) in mathematics shows that if we define a significance limit as 1 %, then students in schools 1–3 improved their mathematics grades. When allowing for significance of 5 % (or confidence of 95 %), the results are the same, but school 4 is on the limit and might be considered for improving its students' grades as well. The means in the paired-samples statistics show that school 1 students improved their grades by 5.70 points, school 2 students improved their grades by 5.08 points, school 3 students improved their grades by 4.56 points, and school 4 students improved their grades by 1.82 points. The differences between the schools were significant.[6]

We also found, using GLM, that there was no statistically significant difference among schools regarding mathematics pre-grades.

We analyzed the achievements of physics as well. The results were similar. For other topics (e.g., science, chemistry, and biology), there were not enough students for obtaining statistically significant results.

[6] Unless otherwise stated, significance in the statistical analyses was set to 5 %.

6 Results in the context of time

Since the time we designed CORDEP, two other teachers (a and b in Table 13) have used the approach to teach robotics courses in high schools and participated in international robotics competitions. They chose to participate in the RoboCup Junior Dance competition[7] whose goal is to compete on 2-min dance performance with emphasis on creativity, entertainment, construction, programming, and reliability. The results of the competition are based on the robot performance and an interview with the robot team that assesses the robot design and the team knowledge about it.

One of the teachers taught a class in 2007 and 2008. Her teams participated in the competition and won in both years. Another teacher, from another school that was trained later, participated in the same competition in 2010 and 2013 and her team won. In addition, the team that won in 2013 also received a special award for electronics design. In 2009, other teachers from these schools taught the robotics course without being trained on the approach; their teams did not win the competition. In 2011, no Israeli team participated in these competitions, and in 2012, these teachers did not participate for various reasons including cost of travel. We see that whenever these teachers participated, they won the competition.

While the results of the competitions are only one item in the overall evaluation of the method, they provide a long-term evaluation of the development process that complements the in-depth analysis conducted over a period of 3 years of which one is presented in this paper. The results in each of these years replicate those presented here. We have many other examples of successful uses of the design methods that are included in CORDEP, but we do not describe them due to space limitations. Nevertheless, we provide one recent example related to the RoboCup competition.[8] One team of university students participated for the first time in the Kid Size League (KSL) that took place in Mexico City on June 2012. One problem they needed to solve was how to make fast convergence of the localization algorithm in order to improve the robot function. The students used ASIT and improved the particle filter (Thrun et al. 2006), which is a fundamental tool for performing comprehensive localization.

7 Discussion

Through careful design, implementation, and testing, we developed a development process for robotic systems—

[7] See http://rcj.robocup.org/dance.html.

[8] http://www.robocup2012.org.

Table 13 Ranking achieved by participating teachers that teach CORDEP

Year	Teacher a	Teacher b
2007	1st	Did not participate
2008	1st	Did not participate
2009	Did not participate	Did not participate
2010	Did not participate	1st
2011	Did not participate	Did not participate
2012	Did not participate	Did not participate
2013	Did not participate	1st

CORDEP—that remarkably achieved its stated goals (Reich et al. 2005; Kolberg et al. 2007a, b) (Table 12; Figs. 10, 11). We described the process of deriving CORDEP and the importance of context-dependent design. In this case, we made adaptations related to the context of the design, namely high school students, high school environment, the product, and the contest. Each of the six design methods had its own special contribution to CORDEP, to the product of the design—the mobile robot—and to the students. The students were aware of the design methods they learned, and we observed that they developed abilities to address the proper design methods to specific problems they encountered in the project.

This is the first time ever that a comprehensive development process has been tested so carefully, including in an international robotics competition, experts' evaluation, analysis of methods' contribution, and students' and teachers' evaluation, and proved successful with such conclusive results. We believe that with the same design approach, technology courses can be taught in universities and industry, yielding even more profound benefits to designers as they would be taught fully without simplifications. Development processes that have been tested in less profound manner might be less valid. This is rarely acknowledged when new methods or processes are presented with little supporting evidence.

Now, we will analyze the results related to the ranking of the robot performances. The robot $ff that was designed by the team members who learned the complete CORDEP won the first place. The difference between the scores of the first and the second places was significant. The score of the first place was even significantly higher than the score of the first place team in the senior division. Further, in the qualification stage, the robot $ff qualified early in the morning in its first out of three runs. A remarkable example of fast problem solving was using the creative thinking method to solve the uneven floor item problem on the qualification day. The speed of implementing the solution (2 h) including hardware and software modifications and calibration runs is a consequence of good design where the design methods allowed for a robust and fast fixable robot. The fact that all other teams had difficulties with the uneven floor indicates that the other teams could not find a good solution for this problem.

The experts' evaluation of the robot performances based on actual tests and videos from the contest clearly distinguishes between the robots that were built by the student teams from different schools. It is obvious from Fig. 10 that the robot of the students that learned the complete CORDEP achieved the highest score in each category in relation to the other robots where the students learned only some, or none, of the design methods. The total average scores further demonstrate the success of the full CORDEP in substantially contributing to the students acquired design skills.

The robots developed by teams that learned CORDEP just partially achieved results inferior to the best robot (although those that studied part of CORDEP competed well with other robots). This is also reflected in their relative inability to deal with the difficult operating modes. All the robots were not tethered and had sound activation for their run start. These are less difficult modes in comparison with obstacle avoidance, overcoming an uneven floor, and returning to home position. Only one additional robot (of school 2) out of the four succeeded in one of its trials to return to its home position. We note that the success of the robots correlates with the number of design methods learned. This suggests that CORDEP as a whole is significant for achieving the best robot performance. Perhaps the robot of school 2 was placed sixth, and not second, due to lack of learning three out of the six methods. The robot of school 3 took 14th place perhaps because of learning only one out of the six methods and was close to the place of the robot from school 4 (16th place) where its team members did not learn the process at all. This pattern of the robot performances repeated itself in the qualification runs and was noticeable in the experts' evaluation of the robot performances.

It is not obvious that the scores of the robot designs will follow those of robot performances. It might happen that an inferior designed robot will have good performances in a specific time and not in other times. In our case however, the robot performances were compatible with their design level. Their scores make a clear distinction between the robot designs made by teams from different schools.

We expected to see a substantial improvement in the students' engineering design skills. Notwithstanding, we were concerned about the ability of high school students to understand and use these methods for their robot design, even with our simplifications. We were surprised how well the students actually learned the design methods and implemented them completely in their design. We also found that when the students saw that one method actually worked and helped them in solving a problem, they became motivated to use these methods until they mastered them.

The complete CORDEP had a strong impact on the students' design quality. Each design method had an essential role in the design and could not be omitted. It showed up in the experts' scores for the students' design. We may say that indeed, there is a correlation between CORDEP and the students' engineering design skills and the students that learned the full CORDEP acquired the highest level of engineering design skills.

The students in their interviews explained the contribution of each one of the six design methods to their design and to the robot performances. From the students' interviews, in particular, from their precise elaborate description of the value of each method, we can conclude that they gained significant understanding of what design is. This further strengthens our belief about the necessity of teaching all the design methods.

In the process of turning inexperienced high school students into designers, the students underwent a mental shift that made them flexible and dynamic problem solvers, possessing a system view, and aware of the design process on the one hand and the details of the product on the other hand.

We decided to examine the student achievements in the science subjects after the students themselves told us from time to time that they felt that they learned better and understood physics or mathematics better. We also received this feedback from their science teachers. One physics teacher that moved to teach in a different school near her house came and asked specifically to start a robotics program in her school because she realized that the students participating in the robotics project have better understanding of what she was teaching.

The analysis of mathematics and physics grades presented in the "Results" section shows that the grades of the students from school 1 improved significantly, in both absolute difference and relative to the other schools. We believe that CORDEP trained the students to acquire systematic attitude toward learning, designing and implementing projects, solving problems creatively, and overcoming difficulties that beforehand were often badly solved or left unsolved.

The PATT scores of school 1 students showed greater improvements in their attitudes relative to the other schools. We learn from the findings that there were significant improvements in all the aspects of the student attitudes. The major improvement was reducing the students' fear from the difficulty of technology. The scores show also the desire of the students to have careers in technology in the future and their interest in technology. We feel that there is a need to discuss further with the students the consequences of technology in order that they have better understanding of this issue.

8 Conclusions

We showed that a new integrated development process for robotic systems was successful. The process provided the students with better tools to deal with many open-ended problems, work in teams, and complete their projects. With the new CORDEP, totally inexperienced students succeeded to create excellent products. The quality of the final product, the mobile robot, designed by the team that used the full CORDEP was higher than the same products designed by other teams and even by engineers.

We found that the students that learned the full CORDEP:

1. created robots with better performances (Table 12),
2. had the best design skills (Figs. 10, 11),
3. improved to a larger extent their grades in mathematics and physics, and
4. improved to a larger extent their attitudes toward engineering/technology.

Since the focus of this paper is not on the educational goals, we did not present supporting evidence for the last two results. The design of a technology course we presented could transform technology education from being frustrating and dull to a lively and worthwhile endeavor that engages students and transforms their learning experience. We believe that such courses could have a major effect on students beyond succeeding in the course itself. It is reasonable to assume that with the same development process, technology courses might be taught at universities and to industry, yielding probably even more profound benefits to designers.

We demonstrated that complex research involving methods, training, and people over extended time could be evaluated carefully using integrative research approach. We showed that results obtained with different evaluations strengthen each other and lead to robust conclusions.

References

Abramovici M, Breuer MA, Friedman AD (1990) Digital systems testing and testable design, Revised Printing. IEEE Press, New York

Agrawala AK, Rauscher TG (1976) Foundations of microprogramming. Academic Press, New York

Ahlgren DJ (2001) Fire-fighting robots and first-year engineering design: trinity college experience. In: Proceedings of 2001 IEEE frontiers in education conference, Reno, pp S2E-1–S2E-6

Ahlgren DJ, Verner IM (2002) Integration of a fire-fighting robot contest in multi-level engineering education. In: Proceedings of the ASEE annual conference, Session 2620

Akao Y, Mazur GH (2003) The leading edge in QFD past, present and future. Int J Qual Reliab Manag 20(1):20–35

Antonelli R, Costa L, Eustachio A, Queiroz R, Martins F, Mello A (2013) Development of a methodology focused on the

improvement of both: ergonomics and comfort of commercial vehicle seats, SAE Technical Paper 2013-36-0216
Atkinson AO (1985) Design for manufacturability: computer-integrated design and manufacturing for product development, SAE Technical Paper 851587
Avizienis A (1976) Fault-tolerant systems. IEEE Trans Comp C25(12):1304–1312
Baily C, Yin C (2009) Design for reliability of power electronics modules. Microelectron Reliab 49(9–11):1250–1255
Baranov S (1994) Logic synthesis for control automata. Kluwer, Netherlands
Barrows H (1985) How to design a problem based curriculum for the pre-clinical years. Springer, Berlin
Becker KH, Maunsaiyat S (2002) Using Thai students' attitudes and concepts of technology. J Tech Educ 13(2):6–20
Ben-Hanan U, Reichsfeld O (2008) Thinking mechatronics. Ort Israel, Tel-Aviv
Betzer N (2002) The knowledge constructing process and the design process constructing among high achiever pupils through project-based-learning (PBL), PhD thesis (Hebrew), School of Education, Tel-Aviv Univ, Tel-Aviv
Boser RA, Palmer JD, Daugherty MK (1998) Students attitudes toward technology in selected technology education programs. J Tech Educ 10(1):4–19
Carlson J, Murphy RR (2005) How UGVs physically fail in the field. IEEE Trans Robot 21(3):423–437
Carter WC, Montgomery HC, Preiss RJ, Reinheimer HJ (1964) Design of serviceability features for the IBM system/360. IBM J Res Dev 8(2):115–126
Chen C (2001) Design for the environment: a quality-based model for green product development. Manag Sci 47(2):250–263
Clark ET Jr (1997) Designing and implementing an integrated curriculum: a student-centered approach. Holistic Education Press, Brandon
Cohen R, Lipon MG, Dai MQ, Benhabib B (1992) Conceptual design of a modular robot. J Mech Des 114(1):117–125
Crawley E, Malmqvist J, Ostlund S, Brodeur D (2007) Rethinking engineering education: The CDIO approach. Springer, Berlin
Crow K (2002) Failure modes and effects analysis (FMEA), DRM associates. http://www.npd-solutions.com/fmea.htm
Dahmus JB, Gonzales-Zugasti JP, Otto KN (2001) Modular product architecture. Des Stud 22(5):409–424
Dertien E (2006) Dynamic walking with Dribbel—design and construction of a passivity-based walking robot. IEEE Robot Autom Mag 3:118–122
Diamond RM (1998) Designing and assessing courses and curricula: a practical guide. Jossey-Bass, San-Francisco
Dieter GE (2000) Engineering design: a materials and processing approach. McGraw-Hill, New York
Franke DW (1998) Configuration research and commercial solutions, AI EDAM, Special Issue: Configuration Design, 12(4):295–300, Cambridge University Press, Cambridge
Frey DD, Herder PM, Wijnia Y, Subramanian E, Katsikopoulos KV, Clausing DP (2009) An evaluation of the Pugh controlled convergence method. Res Eng Design 20:41–58
Galster M, Eberlein A, Moussavi M (2007) Atomic requirements for software architecting. In: SEA '07 proceedings of the 11th IASTED international conference on software engineering and applications. ACTA press, pp 143–148
Gemster G, Leenders AAM (2001) How integrating industrial design in the product development process impacts on company performance. J Prod Innov Manag 18(1):28–38
Göransson B, Gulliksen I, Boivie I (2003) The usability design process: integrating user-centered systems design in the software development process, Special Issue: Bridging the Process and Practice Gaps Between Software Engineering and Human Computer Interaction. Softw Process Improv Practice 8(2):111–131
Habib S (ed) (1988) Microprogramming and firmware engineering methods. van Nostrand, New York
Horowitz R (1999) Creative problem solving in engineering design, PhD Thesis, Tel-Aviv University, Faculty of Engineering
Hubka V (1974) Theorie der Maschinensysteme (Theory of Machine Systems). Springer, Berlin
Iwarsson S, Stahl A (2003) Accessibility, usability and universal design—positioning and definition of concepts describing person-environment relationships. Disabil Rehabil 25(2):57–66
Jones JL, Seiger BA, Flynn AM (1999) Mobile Robots: inspiration to implementation, Second edn. A. K. Peters Ltd, Natick
Kitano H (ed) (1998) RoboCup-97: Robot Soccer World Cup 1. Springer, New York
Kolberg E, Reich Y, Levin I (2003) Project-based high school mechatronics course. Int J Eng Educ Spec Issue Mechatronics Eng Educ 19(4):557–562
Kolberg E, Reich Y, Levin I (2005) Transforming Design Education by Design. In Proceedings of IDETC/CIE 2005 and ASME 2005, California, USA
Kolberg E, Reich Y, Levin I (2007a) Design of Design methodology for autonomous robots. In Proceedings of RoboCup 2007 Symposium, Atlanta, USA July 9–10, (CD Proceedings)
Kolberg E, Reich Y, Levin I (2007b) Express engineering change management. In Proceedings of international conference on engineering designs, ICED 07, Paris, France
Lee CC (1990) Fuzzy logic in control systems: fuzzy logic controller II. IEEE Trans Syst Man Cybern 20(2):419–435
Levin I, Karpovsky M (1998) On-line self-checking of microprogram control units, 4-th IEEE international on-line testing workshop, Capri, Compendium of papers, pp 153–159
Levin I, Levit V (1998) Control ware for learning with mobile robots. Comput Sci Educ 8(3):181–196
Levin I, Mioduser D (1996) A multiple-constructs framework for teaching control concepts. IEEE Trans Educ 39(4):488–496
Levin I, Sinelnikov V, Karpovsky M (2001) Synthesis of ASM-based Self-Checking Controllers. In: Proceedings of EUROMICRO IEEE symposium on digital systems design. Architectures, methods, and tools, warsaw, Poland, pp 87–93
Levin I, Kolberg E, Reich Y (2004a) Robot control teaching with a state machine based design method. Int J Eng Educ 20(2):234–243
Levin I, Levit V, Salzer H (2004b) Atomic specifications and controlware design. WSEAS Trans Comput 3(3):774–782
Li CL (1996) A qualitative and heuristic approach to the conceptual design of mechanisms. Eng Appl Artific Intel 9(1):17–32
Mangir TE, Avizienis A (1982) Fault-tolerant design for VLSI: effect of interconnect requirements on yield improvement of VLSI designs. IEEE Trans Comput C31(7):609–616
Martin F (1994) Learning engineering by building a robot, PhD thesis, Department of Media Arts and Sciences, Massachusetts Institute of Technology
Morag I (2003) Ergonomics in design: The quarterly of human factors applications HFES 11: 6–11, published by SAGE
Mukherjee SS, Kontz M, Reinhardt SK (2002) Detailed design and evaluation of redundant multi-threading alternatives. In: Proceedings 29th annual international symposium on computer architecture, pp 99–110
Myung S, Han S (2001) Knowledge-based parametric design of mechanical products based on configuration design method. Expert Syst Appl 21(2):99–107
Nelson VP (1990) Fault-tolerant computing: fundamental concepts. Computer 23(7):19–25
Pitchumani V (2005) Design for manufacturability. In: Proceedings of design automation conference, Asia and South Pacific

Prasad B (1996) Concurrent engineering fundamentals: integrated product and process organization, vol 1. Prentice Hall, Upper Saddle River

Pugh S (1991) Total design: integrated methods for successful product engineering. Addison-Wesley, Boston

Reich Y (2008) Preventing breakthroughs from breakdowns. In: Proceedings of the 9th Biennial ASME conference on engineering systems design and analysis ESDA2008, Haifa, Israel

Reich Y (2010) My method is better! Res Eng Design 21(3):137–142

Reich Y, Kolberg E, Levin I (2005) Designing designers. In: Proceedings of international conference on engineering design, ICED 05, Melbourne

Reich Y, Kolberg E, Levin I (2006) Designing contexts for learning design. Int J Eng Educ 22(3):489–495

Rhee SJ, Ishii K (2003) Using cost based FMEA to enhance reliability and serviceability. Adv Eng Inform 17(3–4):179–188

Saaty TL (1980) The analytic hierarchy process. McGraw-Hill, New York

Salzer H, Levin I (2004) Atomic requirements in teaching logic control implementation. Int J Eng Educ 20(1):46–51

Shriver B, Smith B (1998) The anatomy of a high-performance microprocessor: a systems perspective. IEEE Computer Society Press, Los Alamitos

Sklar E, Eguchi A (2005) RoboCupJunior: Four Years Later. In: Nardi D, Riedmiller M, Sammut C, Victor JS (eds): RoboCup 2004: Robot Soccer World Cup VIII. Lecture Notes in Computer Science. Springer, New York, pp 172–183

Subrahmanian E, Konda SL, Levy SN, Reich Y, Westerberg AW, Monarch I (1993) Equations aren't enough: informal modeling in design. AI EDAM 7(4):257–274

Thrun S, Burgard W, Fox D (2006) Probabilistic robotics. MIT Press, Cambridge

Tummala RL, Mukherjee R, Xi N, Aslam D, Dulimarta H, Xiao J, Minor M, Dangi G (2002) Climbing the walls: presenting two underactuated kinematic designs for miniature climbing robots. IEEE Robot Autom Mag 4:10–19

Ullman DG (1992) The mechanical design process. McGraw-Hill, New York

Ulrich K (1995) The role of product architecture in the manufacturing firm. Res Policy 24(3):419–440

Verner I, Waks S, Kolberg E (1997) Upgrading technology towards the status of high school matriculation subject: a case study. J Technol Educ 9(1):64–75

Volk K, Yip WM, Lo TK (2003) Hong Kong Pupils' attitudes toward technology: the impact of design and technology programs. J Tech Educ 15(1):48–63

Vozchikov L (2013) Ergonomics method research vehicle mirrors effectiveness 3D Universal model impact classification, SAE Technical Paper 2013-01-2765

Wang L, Shen W, Xie H, Neelamkavil J, Pardasani A (2002) Collaborative conceptual design—state of the art and future trends. Comput Aided Design 34(13):981–996

Youn BD, Park YH, Choi KK (2003) Hybrid analysis method for reliability-based design optimization. J Mech Des 125(2): 221–232

Zadeh LH (1965) Fuzzy sets and systems. System Theory 1965:29–39

Zhang WH, Beckers P, Fleury C (1995) A unified parametric design approach to structural shape optimization. Int J Numer Meth Eng 38(13):2283–2292

Ziv-Av A, Reich Y (2005) SOS—Subjective objective system for generating optimal product concepts. Des Stud 26(5):509–533