

Iterative Design of Advanced Mobile Robots

François Michaud¹, Dominic Létourneau¹, Éric Beaudry²,
Maxime Fréchette¹, Froduald Kabanza² and Michel Lauria¹

¹Department of Electrical Engineering and Computer Engineering, Université de Sherbrooke, Québec, Canada

²Department of Computer Science, Université de Sherbrooke, Québec, Canada

Integration of hardware, software and decisional components is fundamental in the design of advanced mobile robotic systems capable of performing challenging tasks in unstructured and unpredictable environments. We address such integration challenges following an iterative design strategy, centered on a decisional architecture based on the notion of motivated selection of behavior-producing modules. This architecture evolved over the years from the integration of obstacle avoidance, message reading and touch screen graphical interfaces, to localization and mapping, planning and scheduling, sound source localization, tracking and separation, speech recognition and generation on a custom-made interactive robot. Designed to be a scientific robot reporter, the robot provides understandable and configurable interaction, intention and information in a conference setting, reporting its experiences for on-line and off-line analysis. This paper presents the integration of these capabilities on this robot, revealing interesting new issues in terms of planning and scheduling, coordination of audio/visual/graphical capabilities, and monitoring the uses and impacts of the robot's decisional capabilities in unconstrained operating conditions. This paper also outlines new design requirements for our next design iteration, adding compliance to the locomotion and manipulation capabilities of the platform, and natural interaction through gestures using arms, facial expressions and the robot's pose.

Keywords: integration, decisional architecture, task planning and scheduling, human-robot interaction, iterative design

1. Introduction

The dream of having autonomous machines performing useful tasks in everyday environments, as unstructured and unpredictable as they can be, has motivated for many decades research in robotics and artificial intelligence. However, intelligent and autonomous mobile robots are still in what can be identified as research phases,

i.e., mostly basic research and concept formulation, and some preliminary uses of the technology with attempts in clarifying underlying ideas and generalizing the approach [28]. The challenge in making such a dream become a reality lies in the intrinsic complexities and interdependencies of the necessary components to be integrated in a robot. A mobile robot is a system with structural, locomotive, sensing, actuating, energizing and processing capabilities, which set the resources available to deal with the operating conditions of the environment. Such resources are exploited and coordinated based on decisional processes and higher cognitive functions, according to the intended tasks.

In his Presidential Address at the 2005 AAAI (American Association for Artificial Intelligence) Conference, Ronald Brachman argued that Artificial Intelligence (AI) is a system science that must target working in the wild, messy world [4]. This was the initial goal of AI, which revealed to be too difficult to address with the technology 50 years ago, leading to the creation of more specialized subfields. The same can be said about the Robotics scientific community. The underlying complexities in these disciplines are difficult enough to resolve that a “divide-and-conquer” approach is justified. This explains why only a few initiatives around the world tackle directly this challenge. But those that do always identify new ways of solving the integration issues, outlining guidelines and specifications for new research questions or for new technologies aiming to improve robotic capabilities. Continuous technological progress makes it possible to push the state-of-the-art in

robotics closer to its use in day-to-day applications, and it is by addressing directly the integration challenges that such objective can and will be accomplished.

Designing a mobile robot that must operate in public settings probably addresses the most complete set of issues related to autonomous and interactive mobile robots, with system integration playing a fundamental role at all levels. A great variety of robots have been and are currently being designed to operate in natural settings, interacting with humans in the wild, messy world. For the most recent and pertinent related work to our project, we can identify **non-mobile child-size humanoid** robots (e.g., Kaspar, from the RobotCub project, to investigate the use of gestures, expressions, synchronization and imitation; Huggable [31], equipped with full-body sensate skin, silent voice coil-type electromagnetic actuators with integrated position, velocity and force sensing means); **adult-size humanoid torso** (e.g., Dexter [6], for grasping and manipulation using tactile load cells on the finger, investigating decomposition-based planning and knowledge acquisition and programming by demonstration; Domo [12], using stereo vision, force sensing compliant actuators (FSCA) and series elastic actuators (SEA) [27] to investigate alternative approaches to robot manipulation in unstructured conditions); **differential-drive platforms with manipulators** (e.g., B21, PeopleBot and Care-O-Bot platforms equipped usually with one robotic manipulator); **differential-drive mobile torsi** (e.g., Robonaut, from NASA, installed on a Segway Robotic Mobility Platform base and teleoperated; Battlefield Extraction-Assist Robot BEAR from Vecna Robotic-US, using leg-track/dynamic balancing platform to carry fully-weighted human casualties from a teleoperation base); **omnidirectional humanoid base** (e.g., ARMAR III [2], using load cells, torque sensors, tactile skin, stereo vision and six microphone array in the head and a three-tier (planning, coordination, execution) control architecture to accomplish service tasks in a kitchen; HERMES [3], combining stereo vision, kinesthetic, tactile array bumper and single omnidirectional auditory sensing with natural spoken language (input via voice, keyboard or e-mail) with a situation-oriented skill-based architecture).

Even though each of these robots presents interesting functionalities in terms of motion (omnidirectional, articulated torso) and interaction skills (gesture, expressions, tactile, force-controlled actuators, stereo vision, natural language, manipulate objects and collaborate with humans in task-oriented applications), they do not go far in terms of cognition (e.g., autonomous decisions in open settings, adaptation to a variety of situations). They also make compromises on the advanced capabilities of the integrated elements: not all of them use state-of-the-art technologies. These platforms are however necessary in making significant contributions for moving toward higher level of cognition (e.g., future research with Domo targets the implementation of a motivational system with homeostatic balance between basic drives and a cognitive layer, plus incremental learning of sensorimotor experiences [12]).

In opposition, our work initiated from cognitive architectural considerations, and moved to software and hardware requirements of a mobile robot operating in real life settings. EMIB (Emotion and Motivation for Intentional selection and configuration of Behavior-producing modules) was our first computational architecture used to integrate all necessary decisional capabilities of an autonomous robots [22]. It was implemented on a Pioneer 2 robot entry in the 2000 AAI Mobile Robot Challenge. A robot had to start at the entrance of the conference site, find the registration desk, register, perform volunteer duties (e.g., guard an area) and give a presentation [18]. Our team was the first and only one to attempt the event from start to end, using sonars as proximity sensors, navigating in the environment by reading written letters and symbols, using a pan-tilt-zoom (PTZ) camera, interacting with people through a touch-screen interface, displaying a graphical face to express the robot's emotional state, determining what to do next, using a finite-state machine (FSM), recharging itself when needed, and generating a HTML report of its experience [19]. We then started to work on a more advanced platform that we first presented in the 2005 AAI Mobile Robot Competition [21, 20]. Our focus was on integrating perceptual and decisional components addressing all four issues outlined during the AAI events over the years, i.e.: 1) the robust integration of different software packages and intelligent

decision-making capabilities; 2) natural interaction modalities in open settings; 3) adaptation to environmental changes for localization; and 4) monitoring/reporting decisions made by the robot [13, 30, 18, 29]. Trials conducted at the 2005 AAI event helped establish new requirements for providing meaningful and appropriate modalities for reporting and monitoring the robot's states and experiences.

When operating in open settings, interactions are rapid, diverse and context-related. Having an autonomous robot determining on its own when and what it has to do based on a variety of modalities (time constraints, events occurring in the world, requests from users, etc.) also makes it difficult to understand the robot's behavior just by looking at it. Therefore, we concentrated our integration effort in 2006 to design a robot that can interact and explain, through speech and graphical displays, its decisions and its experiences as they occur (for on-line and off-line diagnostics) in open settings. This paper first presents the software/hardware components of our 2006 AAI robot entry, its software and decisional architectures, the technical demonstration made at the conference along with the interfaces developed for reporting the robot's experiences. This outlines the progress made so far in addressing the integration challenge of designing autonomous robots. The paper then describes the design specification of our new advanced mobile robot platform, derived from what we have identified as critical elements in making autonomous systems operating in everyday environments.

2. A Scientific Robot Reporter

Our 2006 AAI robot entry, shown in Figure 1, is a custom-built robot equipped with a SICK LMS200 laser range finder, a Sony SNC-RZ30N 25X pan-tilt-zoom color camera, a Crossbow IMU400CC inertial measurement unit, an array of eight microphones placed in the robot's body, a touch screen interface, an audio system, one on-board computer and two laptop computers. The robot is equipped with a business card dispenser, which is part of the robot's interaction modalities. A small camera (not shown in the picture) was added on the

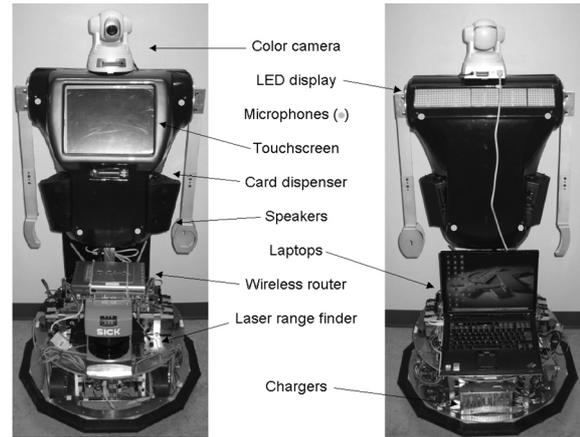


Figure 1. Our 2006 AAI robot entry (front view, back view).

robot's base to allow the robot to detect the presence of electric outlets. The technique used is based on a cascade of boosted classifiers working with Haar-like features [17] and a rule-based analysis of the images.

Figure 2 illustrates the robot's software architecture. It integrates Player for sensor and actuator abstraction layer [34] and CARMEN (Carnegie Mellon Robot Navigation Toolkit) for path planning and localization [24]. Also integrated but not shown in the Figure is Stage/Gazebo for 2D and 3D simulators, and Pmap library¹ for 2D mapping, all designed at the University of Southern California. RobotFlow and FlowDesigner (FD) [8] are also used to implement the behavior-producing modules, the vision mod-

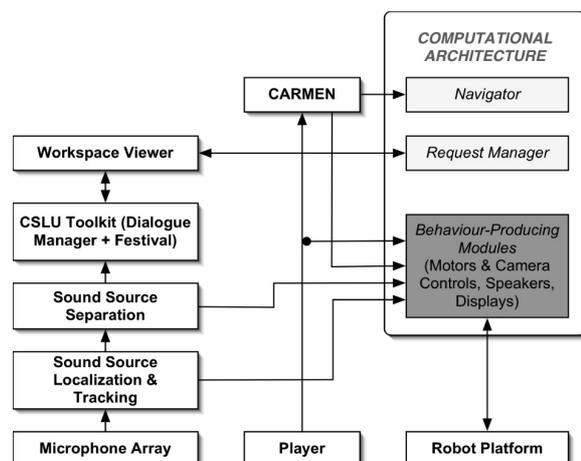


Figure 2. Our robot's software architecture.

¹ <http://robotics.usc.edu/ahoward/pmap/>

ules for reading messages [16] and SSLTS, our real-time sound source localization, tracking [32] and separation [33] system. For speech recognition and dialogue management, we interfaced the CSLU toolkit (<http://cslu.cse.ogi.edu/toolkit/>). One key feature of CSLU is that each grammar can be compiled at run-time, and it is possible to use dynamic grammars that can change on-the-fly before or while the dialogue system is running. Software integration of all these components are made possible using MARIE, a middleware framework oriented towards developing and integrating new and existing software for robotic systems [10, 9].

2.1. Decisional Architecture and Modules

For our participation in the AAI 2006 Mobile Robot Competition, our robot is designed to be a scientific robot reporter, in the sense of a human-robot interaction research assistant. The objective is to have our robot provide understandable and configurable interaction, intention and information in unconstrained environmental conditions, reporting the robot experiences for scientific data analysis.

The robot is programmed to respond to requests from people, and with only one intrinsic goal of wanting to recharge when its energy is getting low (by either going to an outlet identified on the map or by searching for one, and then asking to be plugged in). Our robot's duty is to address these requests to the best of its capabilities and what is 'robotically' possible. Such requests may be to deliver a written or a vocal message to a specific location or to a specific person, to meet at a specific time and place, to schmooze, etc. The robot may receive multiple requests at different periods, and has to autonomously manage what, when and how it can satisfy them.

With our robot, we assume that the most appropriate approach for making a robot navigate in the world and accomplish various tasks is done through the use of behavior-producing modules (also known as behavior-based systems [1]). As a result, representation and abstract reasoning working on top of these modules become a requirement, with the objective of being able to anticipate the effects of decisions while still adapting to the inherently complex properties

of open and unconstrained conditions of natural living environments.

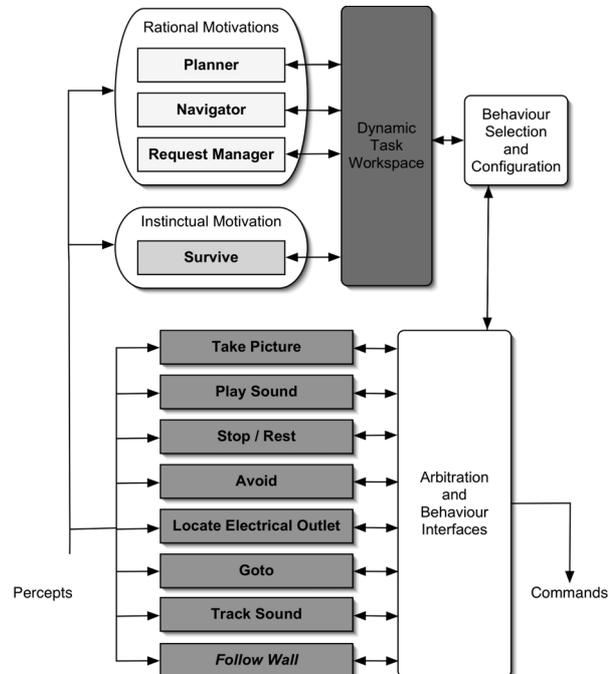


Figure 3. Our robot's decisional architecture.

The decisional architecture developed for our robot's decision-making capabilities is shown in Figure 3. It is based on the notion of having motivated selection of behavior-producing modules. We refer to it as MBA, for Motivated Behavioral Architecture [21, 20]. It is composed of three principal components:

1. Behavior-producing modules (BPMs) define how particular percepts and conditions influence the control of the robot's actuators. Their name represents their purpose. The actual use of a BPM is determined by an arbitration scheme realized through Arbitration and Behaviour Interfaces module (implementing in our case a priority-based / subsumption arbitration), and the BPM activation conditions derived by the Behaviour Selection and Configuration module (explained later).

2. Motivational sources (or Motivations, serving to make the robot do certain tasks) recommend the use or the inhibition of tasks to be accomplished by the robot. Motivational sources are categorized as either instinctual, rational or emotional. This is similar to considering that the human mind is a “committee of the minds,” with instinctual, rational, emotional, etc. minds competing and interacting [35]. Instinctual motivations provide basic operation of the robot using simple rules. Rational motivations are more related to cognitive processes, such as navigation and planning. Emotional motivations monitor conflictual or transitional situations between tasks. The importance of each of these influences explains why manifested behavior can be more influenced by direct perception, by reasoning or by managing commitments and choices. By distributing motivational sources and distinguishing their roles, it is possible to exploit more efficiently the strengths of various influences regarding the tasks the robot must accomplish, while not having to rely on only one of them for the robot to work. This is similar in essence to the basic principles of behavior-based control, but at a higher abstraction level.
3. Dynamic Task Workspace (DTW) serves as the interface between motivational sources and BPMs. Through the DTW, motivations exchange information asynchronously on how to activate, configure and monitor BPMs. The interface between the BPMs and the motivational sources is done through tasks in the DTW. Tasks are data structures that are associated with particular configuration and activation of one or multiple behaviors. The DTW organizes tasks in a tree-like structure according to their interdependencies, from high-level/abstract tasks (e.g., ‘Deliver message’), to primitive/BPM-related tasks (e.g., ‘Avoid’). Motivations can add and modify tasks by submitting modification requests, queries or subscribe to events regarding the task’s status.

By exchanging information through the DTW, motivations are kept generic and independent from each other, allowing motivations to distributively come up with behavior configura-

tion based on the capabilities available to the robot. For instance, one instinctual motivational source may monitor the robot’s energy level to issue a recharging task in the Dynamic Task Workspace, which activates a ‘Locate Electrical Outlet’ behavior that would make the robot detect and dock in a charging station. Meanwhile, if the robot knows where it is and can determine a path to a nearby charging station, a path planning rational motivation can add a subtask of navigating to this position, using a ‘Goto’ behavior. Otherwise, the ‘Locate Electrical Outlet’ behavior will at least allow the robot to recharge opportunistically, when the robot perceives a charging station nearby.

Only instinctual and rational motivations are considered in this study, with rational motivations having greater priority over instinctual ones in case of conflicts. *Survive* makes the robot move safely in the world while monitoring its energy level. *Planner* determines which primitive tasks and which sequence of these tasks are necessary to accomplish higher-level tasks under temporal constraints and the robot’s capabilities (as defined by BPMs). This motivational source is detailed in Section 2.1.1. *Navigator* determines the path to a specific location to accomplish tasks posted in the DTW. *Request Manager* handles task requested by the users via the touch screen interface or from vocal commands. Task requests are then processed by *Planner* and added to the actual plan, if appropriate.

With multiple tasks being issued by the motivational sources, the Behavior Selection and Configuration module determines which behaviors are to be activated according to recommendations made by motivational sources, with or without particular configuration (e.g., a destination to go to). A recommendation can either be negative, neutral or positive, or take on real values within this range regarding the desirability of the robot to accomplish specific tasks. Activation values reflect the resulting robot’s intentions derived from interactions between the motivational sources. Behavior use and other information coming from behavior and that can be useful for task representation and monitoring are also communicated through the Behavior Selection and Configuration module.

2.1.1. Task Planning Motivation

This motivational source invokes, when a new task is placed in the DTW, a planning algorithm that combines principles from SHOP2 HTN planning algorithm [25] and SAPA [11]. As in SHOP2, we specify a planning domain (e.g., the domain of attending a conference at AAI) by describing the robot primitive behaviors in terms of template tasks and methods for recursively decomposing template tasks down to primitive tasks which are atomic actions. For instance, we can specify that the task of making a presentation at location px is from time t_1 to time t_2 , and that it can be decomposed into the simpler subtasks of going to px and presenting at time t_1 . Decomposition of tasks into simpler ones is given with preconditions under which the decomposition is logically sound and time constraints for ensuring its consistency. Given such a set of task specifications, the planning algorithm consists of searching through the space of tasks and world states.

More specifically, starting from a given set of initial tasks and a current state describing the robot state and environment situation, the planner explores a space of nodes where each node represents: the current list of subtasks; the current robot state and environment situation; the current plan (initially empty). A current node is expanded into successors by decomposing one of the current task into simpler ones (using the specified task decomposition method) or by validating a current primitive task against the current world state (this is done by checking its precondition and updating the current world state using the primitive task's effects) and adding it to the current plan. The search process stops when a node is reached with no more task to decompose. On a node, there can be different ways of decomposing a task and different orders in which to decompose them; backtracking is invoked to consider the different alternatives until obtaining a solution. This can be a source of exponential blow up during the search process. By carefully engineering the decomposition methods to convey some search control strategy, it is possible to limit this state explosion [25].

Task Planning with Temporal Constraints. A normal HTN planning algorithm does not con-

sider temporal constraints [25, 26]. To implement this, we modified the basic HTN planning algorithm by: (a) adding a *current-time* variable into the representation of a current state during search; (b) specifying time constraints in the specification based on this variable in the precondition of task decomposition methods and of primitive tasks; (c) allowing the specification of conditional effect update for primitive tasks based on this variable. That way, we extend the HTN concept to support time-constrained tasks by adding temporal constraints at each decomposition level. The planner can use these temporal constraints to add partial orders on tasks. These partial orders reduce the search space and accelerate the plan generation process. These temporal constraints can also be transferred when a high-level task is decomposed into lower-level tasks.

When defining temporal intervals in the domain specification, care must be taken to establish a good compromise between safety and efficiency for task execution. Being too optimistic may cause plan failure because of lack of time. For instance, assuming that the robot can navigate at high speed from one location to the other will cause a plan to fail if unpredictable events slow down the robot. To solve this problem, the solution is to be conservative in the domain model and assume the worst case scenario. On the other hand, being too conservative may lead to no solution. Therefore, temporal intervals in the domain model are specified using an average speed (0.14 m/s) lower than the real average speed (0.19 m/s) of the robot.

Over the last years, efforts have been made in creating formal techniques for planning under temporal and resource uncertainty [5]. To keep our approach simple and to avoid having to deal with complex models of action duration, we use the following heuristics: planning is initiated first using a conservative action duration model and, when a possible opportunity (using the temporal information associated with the tasks) is detected, the planner is reinvoked to find a better plan. In other words, if the robot proceeds faster than it is planned (i.e., the end of the updated projected action is lower than the end of the planned action), it is possible that a better plan exists, and replanning therefore occurs.

Time Windowing. Using a technique borrowed from SAPA [11], our planner post-process the plan generated by the search process to obtain a plan with time windowing for actions. During search, each node is associated with a fixed time stamp (that is, the *current – time* variable), and at the end of the search process we obtain a plan consisting of a sequence of actions each assigned with a time stamp indicating when it should be executed. This sequence of actions is post-processed based on time constraints in the domain specification (i.e., constraints attached to task decomposition methods and primitive tasks) to derive time intervals within which actions can be executed without jeopardizing the correctness of the plan.

Task Filtering and Priority Handling In a traditional planning setting, a list of initial tasks is considered as a conjunctive goals. If the planner fails to find a plan that achieves them all, it reports failure. In the context of the AAI Challenge as well as in many real life situations, we expect the robot to accomplish as many tasks as possible, with some given preferences among the tasks. For instance, the robot may fail to deliver one message at the right place, but successfully delivers another one. This would be acceptable depending on the environment conditions.

We implement a robot mission as a list of tasks, each associated with a degree of preference or priority level. Then we iteratively use the HTN planner to obtain a plan for a series of approximation of the mission, with decreasing level of accuracy. Specifically, initially we call the planner with the entire mission; if it fails to compute a plan within a deadline set empirically in the robot architecture (typically 1 second), a lowest priority task is removed from the mission and the planner is called with the remaining mission. This is repeated until a solution plan is found. If the mission becomes empty before a solution is found, failure is returned (the entire mission is impossible). Clearly, this model can be easily configured to include vital tasks that cause failure whenever any of them is not achievable. We also use some straightforward static analysis of the mission to exclude for instance low priority tasks that apparently conflict with higher priority ones.

Anytime Task Planning. Similarly to SAPA [11] and SHOP2 [25], our planner has an anytime planning capability, which is important in mobile robotic applications. When a plan is found, the planner tries to optimize it by testing alternative plans for the same mission until the empirical maximum planning time is reached.

On-line Task Planning. The integration of planning capabilities into a robotic system must take into account real-time constraints and unexpected external events that conflict with previously generated plans. To reduce processing delays in the system, our planner is implemented as a library. The MARIE application adapter that links the planner to the rest of the computational architecture loads the planner library, its domain and world representation (e.g., operators, preconditions and effects) at initialization. The planner remains in the memory and does not have to be loaded each time it is invoked. A navigation table (providing the distances from each pairs of waypoints) also remains in the memory and is dynamically updated during the mission. External events are accounted for by monitoring the execution of a plan and by validating preconditions and time constraints of remaining actions in the plan. Violated preconditions activates re-planning.

The robot can receive task requests at any time during a mission. This requires the robot to update its plan even if it is not at a specified waypoint in its world representation. When generating a task plan, the duration of going from one waypoint to another is instantiated dynamically based on the current environment. That is, the duration of a *Goto(X)* action is not taken from the navigation table, but is rather estimated from the actual distance to the targeted waypoint *X*, as provided by an internal path-planning algorithm. Therefore, *Planner* only has to deal with high-level navigation tasks, and it is not necessary to plan intermediate waypoints between two waypoints that are not directly connected. To estimate the duration for navigating from a waypoint to another, the planner uses a navigation table of the size n^2 where n is the number of defined waypoints. This table is initialized by computing the minimum distance between waypoints with a classic A* algorithm. Each time a new waypoint is added on the map, the distances table is updated dynamically.

2.1.2. Interaction Modalities

With a distributed cognitive architecture integrating a high number of capabilities to be used in open conditions, it is difficult to assess what is going on simply by looking at the robot's behavior, or by analyzing log files off-line. More dynamic and interactive tools are required. Here is the set of modalities designed to improve our understanding of integrated modalities (vision, audition, graphical, navigation) on our robot.

- Visualization tools of decisional components.** The underlying objectives of such tools is to facilitate integration work by offering visualization tools of the decisional components added to the robot. It requires coherent and user-friendly representation of the integrated components so that developers can concentrate on their combined usage instead of spending time extracting and interpreting data in log files. In 2005, we developed a log viewing application, allowing us to monitor off-line the complete set of states of the robot through log files created by the different software components [21]. In 2006, we extended this tool to provide a dynamic and real-time view of what the robot is actually doing and planning to do, on-line. We call this updated version the WorkspaceViewer. With the WorkspaceViewer, it is now possible to display, directly

on our robot's graphical interface, contextual information according to its current plan (e.g., behavioral configuration, map with dynamic places, active tasks). In addition, we can still use the basic off-line log viewing application to replay logs off-line.

Figure 4 shows a representation of the WorkspaceViewer's main window. The upper section contains a timeline view of DTW events (first line), planner events (second line), and behaviors' activations and exploitations (third line). The bottom section shows detailed information according to the position of the vertical marker on the timeline: a list of DTW's tasks and properties (first window from the left), active motivations (second window), the current plan (third window), the map of the environment and the trajectory of the robot (fourth window), the behaviors and their activations and exploitations (under the first window). The WorkspaceViewer is directly connected to the DTW and displays information as they become available in real-time. The WorkspaceViewer is also linked to *Request Manager* motivation to handle user requests.

- Visualization of the SSLTS results.** Audio data is difficult to monitor in dynamic and open conditions: it is not persistent as a

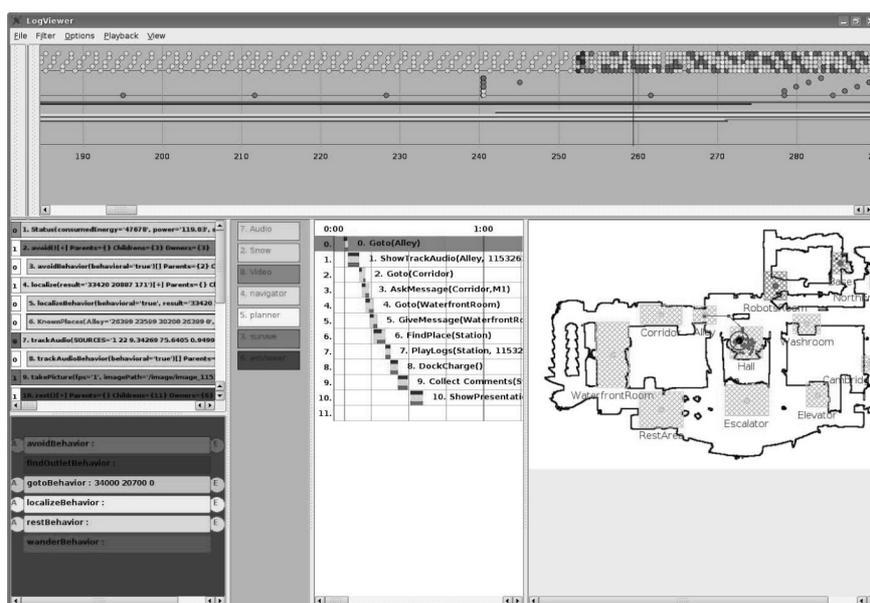


Figure 4. WorkspaceViewer's main window.

visual feature can be, for instance. An interface representing where the sound sources around the robot are detected and what was recorded by the robot revealed to be necessary from what we experienced during our 2005 participation. Figure 5 illustrates the interface developed. The upper part shows the angle of the perceived sound sources around the robot, in relation to time. The interface shows in real-time the sound sources perceived, using dots of a distinct color. The sound sources are saved, and the user can select them and play back what the SSLTS generated. This interface reveals to be very valuable for explaining what the robot can hear, to construct a database of typical audio streams in a particular setting, and to analyze the performance of the entire audio module (allowing to diagnose potential difficulties that the speech recognition module has to face).

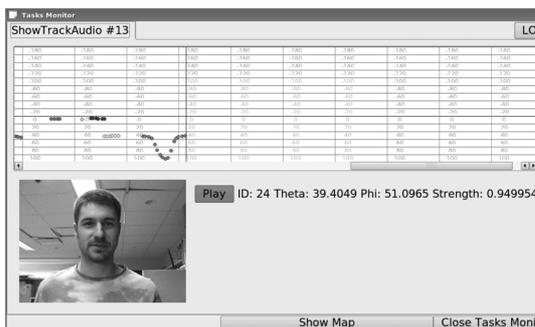


Figure 5. Track audio interface.

For instance, more than 6600 audio streams were recorded over the twelve hours of operation of the robot in open settings at the AAI 2006 conference, held at Seaport Hotel and World Trade Center in Boston. From this set of audio streams, we observed that around 61% were audio content (people talking nearby the robot, sounds of different sort) adequately processed by our SSLTS system (i.e., the audio streams were correctly separated and located, and are understandable by a human listener). However, they were not related to specific interaction with the robot, and therefore revealed to be useless for interacting with our robot. 3.5% of the audio streams were inaudible by a human listener, and an additional 34%

were either very short (e.g., caused by a noise burst or laughter) or truncated (e.g., when the robot starts talking, to avoid making the robot hear itself talk; sound amplitude too low). Implementation problems caused these limitations, as we diagnosed after the event. This left around 1.5% of audio streams intended for specific interaction with the robot adequately processed by the SSLTS system. All were correctly processed using CLSU in specific interaction our robot had with human interlocutors. This clearly demonstrates the challenge of making an artificial audition system for operation in open conditions. Such auditory capability working in real-time in such conditions has not yet been demonstrated on other robots. The performance observed is impressive, but the challenges to overcome to make it more robust and efficient are still quite important. We will continue to improve SSLTS' performance by not conducting speech recognition on audio streams that are too small (i.e., < 1 sec) and by making it possible to concurrently run multiple speech recognition processes (currently only one stream can be processed at a time).

- **Multimodal interaction interfaces.** It is sometimes difficult to know exactly which task is prioritized by the robot at a given time, making the interlocutors unable to know what the robot is actually doing and evaluate its performance. So we made the robot indicate its intention verbally and also graphically (in case, for whatever reasons, the in-

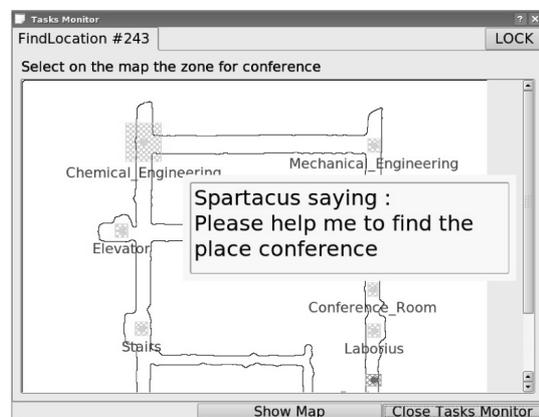


Figure 6. Graphical display for an assistance request to find a location (in laboratory conditions).

terlocutor is not able to understand the message). Figure 6 illustrates the graphical interface with which our robot is requesting assistance to find a location in the convention center.

The graphical interfaces of the robot were made so that users can indicate through push buttons (using the touch screen) what they want our robot to do. We also made it possible for the users to make these requests verbally, saying out loud the names of the buttons. To facilitate the recognition process, specific grammar and dialogue managers are loaded in CSLU for each graphical mode. The audio and graphical interaction are therefore tightly integrated together.

2.2. Demonstration and Results

Our robot demonstrated its capabilities in the AAI 2006 Human-Robot Interaction Event, evaluated in a technical and a public demonstrations. Our robot entered five of the seven categories by doing the following:

- Natural Language Understanding and Action Execution: Following requests from humans, written or verbal, for making the robot do different tasks.
- Perceptual Learning Through Human Teaching: Learning of a location, specified by humans or identified by the robot (i.e., electrical outlet), and being able to remember it.
- Perception, Reasoning, and Action: Temporal reasoning, planning and scheduling of tasks and intents.
- Shared Attention, Common Workspace, Intent Detection: Directing camera view in the direction of the speaker, communicate temporal constraints regarding task, and use of the dynamic (context-configured) viewer, on-line and off-line.
- Integration Challenge Categories 3 to 6: Robot Scientific Reporter/Assistant demo, with understandable interaction, decision and situations experienced in unconstrained conditions.

Our technical demonstration, programmed to last 25 minutes (setting an overall fixed time

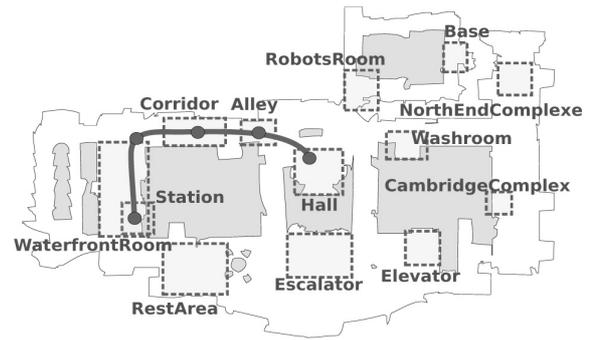


Figure 7. Our robot's intended trajectory for the technical presentation.

constraint for the mission), consisted of five phases done in the area represented in Figure 7:

1. INTRODUCTION. Presentation (max 4:30 min) at *Alley* of our robot's features: track audio viewer for separated sources; directing camera view toward the speaker, with snapshots memorized every second; graphic displays for user input; context-based grammar; pre-mapping of the area; Gantt chart representation of the plan.
2. NAVIGATION. Go to *Corridor* to receive a message (voice attachment or text) to be delivered at *WaterfontRoom*. No time constraints are specified for this task. A specific graphical interface is displayed by the *WorkspaceViewer* to get the message. The *Planner* inserts this task if time permits between existing tasks (which are time constrained).
3. PLANNED RECHARGE. Initiate a recharging task, asking somebody to plug the robot in an electrical outlet. This should occur no later than 25 minutes after the beginning of the presentation at the *Station*. However, *Station* is an unknown location when the demonstration starts. There are two ways the location can be determined : from user input with the touch screen interface, or automatically added if an electrical outlet is perceived.
4. PLAYBACK. Accelerated replay of what the robot experienced during the demonstration. The *WorkspaceViewer* program starts an automatic replay from the beginning of the demonstration, and stops at important events. Judges can then see a snapshot of the active behaviors, the plan, the trajectory

of the robot and active tasks. Playback occurs 15 minutes after the beginning of the demonstration at *Station* (max 2:00 min).

5. **QUESTIONS AND COMMENTS.** Ask judges to enter comments regarding its performance and capabilities through a *WorkspaceViewer* contextual interface. Judges could enter textual or vocal comments if they want. This task occurs 20 minutes after the beginning of the presentation (max 4 min). One minute before the end, a ‘Thank you’ message is displayed on the screen, and our robot gives one business card.

Figure 8 shows the dry run conducted minutes before the technical presentation, without the jury members. We started our robot in the *Hall* with a mission making it go through *Alley*, *Corridor*, *Waterfront Room*, to finally end its presentation at *Station*. Everything went smoothly and worked out fine.

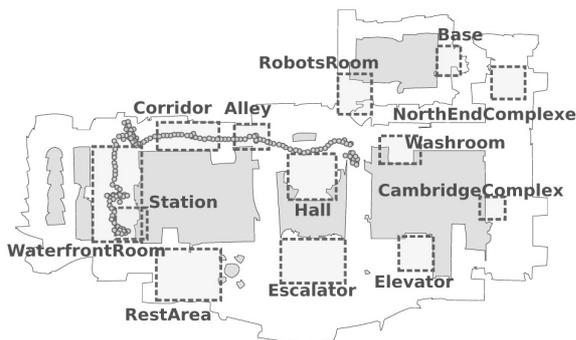


Figure 8. Localization results with nobody around our robot.

Figure 9 shows what actually occurred during the technical demonstration. Compared to Figure 8, our robot had a lot of difficulties localizing its position when people and judges were around the robot. CARMEN using laser range finder data had difficulties finding reference points to its internal map. The estimated positions, represented with dots in Figure 9, are scattered and even sometimes outside the map. Additional fine-tuning of parameters of probabilistic odometry model in CARMEN configuration file would have been required to improve localization performance in crowded conditions. We had to reposition manually the robot on the map when this happened, and our robot arrived two minutes late at *Alley*. With our

robot already following a tight schedule, it had a lot of trouble going through all of the planned tasks: they were dismissed in sequence to try to catch up with the schedule. Most of the time, we had to manually demonstrate the capabilities by modifying the initial plan and removing tasks so that the time constraints could fit in a new plan. This went on until the end, even exceeding the 25 minutes time constraint. Therefore, it turned out that localization performance played a huge impact on the overall performance of our robot. We knew that the time constraints were difficult to meet, but we did not expect such poor performance with localization. This revealed the influence of tightly coupling the planner and the localizer, and that multiple localization capabilities may be required when navigating with too many people surrounding the robot, blocking the laser’s field of view to get good position estimates.

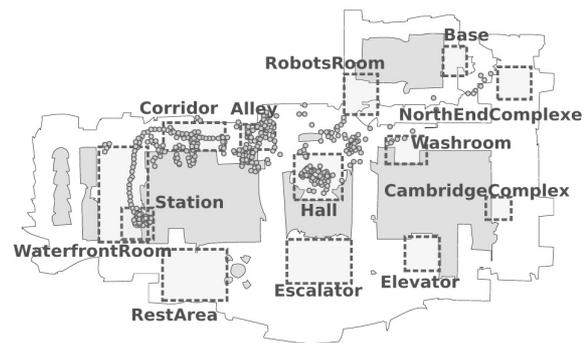


Figure 9. Localization results during the technical presentation.

Our public demonstration was made to be more interactive and entertaining. For instance, we made our robot interact with people by playing games (music or science trivia questions, fortune cookies). In very crowded conditions, positioning the camera in the direction of the speaker worked very well, as did the separation of sources considering the very noisy conditions in which the robot was in. Looking at the log files, it was possible to clearly follow conversations the robot had with people. With the event taking place outside the technical demonstration area, no map was made of the area, and our robot only wandered around, being stopped most of the time to interact with people. With the observed performances during the technical demonstration, having a map would not have helped the robot localize itself in the area (it

was much more crowded than our robot had experienced during the technical demonstration).

Overall, our robot was the overall winner of the event, finishing first in four categories (Perceptual Learning through Human Teaching; Perception, Reasoning and Action; Shared Attention, Common Workspace, Intent Detection; Integration Challenge), and second in one (Natural Language Understanding and Action Execution). It was also awarded the Public Choice Award.

3. Next Iteration in Designing Advanced Mobile Robots

Our 2006 implementation showed increased reliability and capabilities of MBA (especially the DTW) in a stable and integrated implementation of the components described in Section 2. Our planner is capable of dealing with conditions temporal constraints violation, opportunity detection, task cancellation, unknown waypoint that would occur in real life settings and within a computational architecture using distributed modules (compared to architectures with one centralized module for goal/task generation). Our robot is equipped with a dynamic viewer with voice and graphical interaction, contextualized based on the robot's active tasks.

However, it is only one step in coming up with the most efficient integration of these modalities. We now have the necessary tools for easy configuration of robot missions and for on- and off-line analysis of experienced situations (trajectories, requests, sound sources, pictures, internal states, etc.), useful in human-robot interaction experiments and AI algorithms in unconstrained conditions. In future work, we want to use these tools to address different research issues, such as the influence of directed attention in vocal interaction, the preferred way and time spent interacting with a robot, the most efficient strategy to navigate in a crowd or to find an unknown location, to schmooze and make acquaintance with people, and to compare different components for the robot's modalities (e.g., speech recognition software such as NUANCE and Sphinx, dialogue software such as CLSU and Collagen, SLAM algorithm such as CAR-MEN and vSLAM).

Still, we must remember that the intelligence manifested by a machine is limited by its physical, perceptual and reasoning capabilities. Using our robot entries to the AAAI 2000, 2005 and 2006 events, we were able to study the added value of integrating hardware and software components, energetic versus weight versus power considerations, navigation capabilities, artificial audition and vision, task planning and scheduling, distributed decisional architecture and visualization tools for evaluation in dynamic conditions. The software and decisional architectures can be ported and adapted to another platform, but their strengths and limitations can only be adequately characterized when used on platforms with greater capabilities. But we need to add significant improvements to the platform to make even more progress. Therefore, from what we observed from our trials with our robots and in line with the current state-of-the-art in mobile robotic development, we have identified and are working on two key innovations that would greatly benefit our objective of designing advanced mobile robots operating in everyday environments and interacting with humans.

- **Compliance.** A robot mechanically interacts with its environment when the dynamics of the coupled system (the robot and environment in contact) differs significantly from the dynamics of the robot alone [7,14]. Not all robots mechanically interact with their environment. For instance, the vast majority of robots used for industrial applications do not interact significantly. In many cases, a limited amount of physical interaction can be tolerated without specific modeling or control. However, for complex robotic tasks (e.g., manipulation, locomotion), the lack of knowledge of precise interaction models, the difficulties to precisely measure the task associated physical quantities (e.g., position of contact points, interaction forces) in real-time, the finite sampling time of digital control loops and the non-collocation of sensors and transducers have negative effects on performance and stability of robots when using simple force or simple movement controllers. For robots to take a more significant role in human life, they must safely manage intentional and unintentional physical contact with humans, even while performing high-force tasks. These

challenges occur when interacting with the environment produces coupled system dynamics that differ significantly from the dynamics of the robot system alone.

We are currently developing a new compliant actuator named Differential Elastic Actuator (DEA), specifically designed to deal with the lack of knowledge on precise interaction models in natural settings and the difficulties to precisely measure in real-time forces and positions associated to complex robotic tasks like manipulation and locomotion. Compared to FSCA and SEA, DEA [15] uses a differential coupling instead of a serial coupling between a high impedance mechanical speed source and a low impedance mechanical spring. This results in a more compact and simpler solution, with similar performances. DEA are high performance actuators providing higher torque/encumbrance ratio, higher torque/weight ratio, lower rotational inertia, lower rotation velocity of the flexible element, and improved effort transmission through the motors chassis.

- **Gesture in natural interaction.** Natural interaction is mainly influenced by perception through senses like vision, audition and touch, and by actions like speech, graphics and also gesture. Gesture, i.e., a motion made to express or help express thought, was not covered in our past iterations, and would greatly enhance the way the robot communicates with humans. Gesture can be made using arms, facial expression, but also through the pose of the robot (rising itself up or moving down).

With these elements in mind, we are currently designing our third iteration of interactive autonomous mobile robot platform. The platform is shown in Figure 10. The mobile base is a legged-tracked platform capable of changing the orientation of its four articulations. Each articulation has three degrees of freedom (DOF): it can rotate 360 degrees around its point of attachment to the chassis, can change its orientation over 180 degrees, and rotate to propulse the robot. This platform is inspired from the AZ-IMUT platform [23], built using stiff transmission mechanisms, with motors and gearboxes placed directly at the attachment points of the articulations. Such design choices made the

platform vulnerable to shocks when moving over rough terrain, especially with its articulations pointing down: an undetected obstacle touching the tip of an articulation would create an important reaction force at the articulation's attachment point to the chassis and inside his non back drivable gearbox. The new base uses DEA actuators for compliant locomotion control: 4 DEA for the direction of its articulations (making it possible to guide the platform simply through touch); 4 DEA at the attachment point of the leg-tracks (for force control when the robot stands up). This should improve the locomotion capabilities of the platform, making it omnidirectional and intrinsically safe. The humanoid torso is capable of arm gesture and facial expressions. Its arms will use DEA actuators for safe manipulation of objects and interaction with people. All of these elements are currently being fabricated, assembled and tested, and will be integrated by end of 2008.

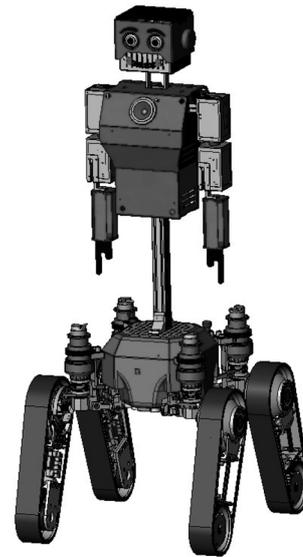


Figure 10. Our third iteration in designing advanced mobile robotic platform.

4. Conclusion

Mobile robotics is one of the greatest domains for systems engineering: it requires the integration of sensors, actuators, energy sources, embedded computing, decision-making algorithms, etc., all into one system. Only technologies and methodologies that work within the interdependent constraints of these elements, addressed in a holistic approach, can be useful.

The integration work is influenced by technological factors as well as by the operating conditions of the robotic platform. With continuing technological progress, designs and tools must take into consideration possible extensions to the integration of modular elements of the system. Since a robot is an embodied system, it has to deal with real world dynamics and issues related to the intended application. Therefore, integration directly influences scientific progress in mobile robots, and it is important to address such challenges by developing and addressing them in designing new platforms and conduct real world field trials. The primary research contribution of this paper is to present what we have done so far with our AAI 2006 entry, and what we are aiming to accomplish with our new design iteration. This puts in perspective what can be done and what remains to be solved.

The design of advanced mobile robotic platforms is still in a basic research phase [28], investigating ideas and concepts, putting initial structure on problems and framing critical research questions. Progress is accomplished through multiple iterations of increasing complexity, identifying new challenges and research issues as we moved through these iterations. What we have identified with our AAI 2006 robot entry at all levels, from structural to hardware and software components, has led to the design specifications of our new robot. It is by designing more advanced platforms that we will be able to measure progress at all these levels.

As a corollary, one of our objectives is to share our efforts by making available our hardware and software components. Advanced mobile robotics is a field so broad that it is not possible to be an expert in all related areas. Collaborative work is therefore a necessary element in our attempts, as a community, to significantly accelerate scientific progress of the field. More specifically, we plan to use our new robot to conduct comparative and integrated evaluations of robotic software capabilities and architectures on a common platform and shared test conditions, thus eliminating any bias to conduct such evaluations. This way, we should be able to move research activities from feasibility studies to comparative evaluations and human-robot interaction field studies, addressing an all new level with clear innovations in terms of motion, interaction and cognition capabilities, both individually and integrated.

5. Acknowledgment

F. Michaud holds the Canada Research Chair (CRC) in Mobile Robotics and Autonomous Intelligent Systems. Support for this work is provided by the Natural Sciences and Engineering Research Council of Canada, the Canada Research Chair program and the Canadian Foundation for Innovation.

References

- [1] ARKIN, R. C. 1998. *Behavior-based Robotics*. The MIT Press.
- [2] ASFOUR, T.; REGENSTEIN, K.; AZAD, P.; SCHRODER, J.; BIERBAUM, A.; VAHRENKAMP, N.; AND DILLMAN, R. 2006. Armar-III: An integrated humanoid platform for sensory-motor control. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robotics*, 169–175.
- [3] BISCHOFF, R., AND GRAEFE, V. 2004. Hermes: A versatile personal assistant robot. *Proceedings of the IEEE Special Issue on Human Interactive Robots for Psychological Enrichment* 1759–1779.
- [4] BRACHMAN, R. J. 2006. (AA)AI more than the sum of its parts. *AI Magazine* 27(4):19–34.
- [5] BRESINA, J.; DEARDEN, R.; MEULEAU, N.; RAMKRISHNAN, S.; SMITH, D.; AND WASHINGTON, R. 2002. Planning under continuous time and resource uncertainty: A challenge for AI. In *Proceedings of the 19th Conference on Uncertainty in AI*, 77–84.
- [6] BROCK, O.; FAGG, A.; GRUPEN, R.; PLATT, R.; ROSENSTEIN, M.; AND SWEENEY, J. 2005. A framework for learning and control in intelligent humanoid robots. *International Journal of Humanoid Robotics* 2(3):301–336.
- [7] BUERGER, S. P. 2005. *Stable, High-force, Low Impedance Robotic Actuators for Human-interactive Machines*. Ph.D. Dissertation, MIT.
- [8] COTE, C.; LETOURNEAU, D.; MICHAUD, F.; VALIN, J.-M.; BROSSEAU, Y.; RAIEVSKY, C.; LEMAY, M.; AND TRAN, V. 2004. Code reusability tools for programming mobile robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1820–1825.
- [9] COTE, C.; BROSSEAU, Y.; LETOURNEAU, D.; RAIEVSKY, C.; AND MICHAUD, F. 2006a. Using MARIE in software development and integration for autonomous mobile robotics. *International Journal of Advanced Robotic Systems, Special Issue on Software Development and Integration in Robotics* 3(1):55–60.

- [10] COTE, C.; LETOURNEAU, D.; RAIIEVSKY, C.; BROSSEAU, Y.; AND MICHAUD, F. 2006b. Using MARIE for mobile robot software development and integration. In Brugali, D., ed., *Software Engineering for Experimental Robotics*. Springer Tracts on Advanced Robotics.
- [11] DO, M., AND KAMBHAMPATI, S. 2003. SAPA: A scalable multi-objective metric temporal planner. *Journal of Artificial Intelligence Research* 20:155–194.
- [12] EDSINGER-GONZALES, A., AND WEBER, J. 2004. Domo: A force sensing humanoid robot for manipulation research. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robotics*, 273–291.
- [13] GOCKLEY, R.; SIMMONS, R.; WANG, J.; BUSQUETS, D.; DISALVO, C.; CAFFREY, K.; ROSENTHAL, S.; MINK, J.; THOMAS, S.; ADAMS, W.; LAUDUCCL, T.; BUGAJSKA, M.; PERZANOWSKI, D.; AND SCHULTZ, A. 2004. Grace and George: Social robots at AAI. Technical Report WS-04-11, AAI Mobile Robot Competition Workshop. pp. 15–20.
- [14] HOGAN, N., AND BUERGER, S. P. 2005. Impedance and interaction control. In *Robotics and Automation Handbook*. CRC Press.
- [15] LAURIA, M.; LEGAULT, M.-A.; LAVOIE, M.-A.; AND MICHAUD, F. 2008. Differential elastic actuator for robotic interaction tasks. Submitted to *IEEE International Conference on Robotics and Automation*.
- [16] LETOURNEAU, D.; MICHAUD, F.; AND VALIN, J.-M. 2004. Autonomous robot that can read. *EURASIP Journal on Applied Signal Processing, Special Issue on Advances in Intelligent Vision Systems: Methods and Applications* 17:1–14.
- [17] LIENHART, R., AND MAYDT, J. 2002. An extended set of Haar-like features for rapid object detection. In *Proceedings of the IEEE International Conference on Image Processing*, volume 1, 900–903.
- [18] MAXWELL, B.; SMART, W.; JACOFF, A.; CASPER, J.; WEISS, B.; SCHOLTZ, J.; YANCO, H.; MICIRE, M.; STROUPE, A.; STORMONT, D.; AND LAUWERS, T. 2004. 2003 AAI robot competition and exhibition. *AI Magazine* 25(2):68–80.
- [19] MICHAUD, F.; AUDET, J.; LETOURNEAU, D.; LUSSIER, L.; THEBERGE-TURMEL, C.; AND CARON, S. 2001. Experiences with an autonomous robot attending the AAI conference. *IEEE Intelligent Systems* 16(5):23–29.
- [20] MICHAUD, F.; BROSSEAU, Y.; COTE, C.; LETOURNEAU, D.; MOISAN, P.; PONCHON, A.; RAIIEVSKY, C.; VALIN, J.-M.; BEAUDRY, E.; AND KABANZA, F. 2005. Modularity and integration in the design of a socially interactive robot. In *Proceedings of the IEEE International Workshop on Robot and Human Interactive Communication*, 172–177.
- [21] MICHAUD, F.; COTE, C.; LETOURNEAU, D.; BROSSEAU, Y.; VALIN, J.-M.; BEAUDRY, E.; RAIIEVSKY, C.; PONCHON, A.; MOISAN, P.; LEPAGE, P.; MORIN, Y.; GAGNON, F.; GIGUERE, P.; ROUX, M.-A.; CARON, S.; FRENETTE, P.; AND F.KABANZA. 2007. Spartacus attending the 2005 AAI conference. *Autonomous Robots, Special Issue on AAI Mobile Robot Competition* 22(4):369–384.
- [22] MICHAUD, F. 2002. EMIB – Computational architecture based on emotion and motivation for intentional selection and configuration of behaviour-producing modules. *Cognitive Science Quarterly, Special Issue on Desires, Goals, Intentions, and Values: Computational Architectures* 3-4:340–361.
- [23] MICHAUD, F.; LETOURNEAU, D.; ARSENAULT, M.; BERGERON, Y.; CADRIN, R.; GAGNON, F.; LEGAULT, M.-A.; MILLETTE, M.; PARE, J.-F.; TREMBLAY, M.-C.; LEPAGE, P.; MORIN, Y.; AND CARON, S. 2005. Multi-modal locomotion robotic platform using leg-track-wheel articulations. *Autonomous Robots, Special Issue on Unconventional Robotic Mobility* 18(2):137–156.
- [24] MONTEMERLO, M.; ROY, N.; AND THRUN, S. 2003. Perspectives on standardization in mobile robot programming: The Carnegie Mellon navigation (CARMEN) toolkit. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2436–2441.
- [25] NAU, D.; AU, T.; ILGHAMI, O.; KUTER, U.; MURDOCK, J.; WU, D.; AND YAMAN, F. 2003. SHOP2: An HTN planning system. *Journal of Artificial Intelligence Research* 20:379–404.
- [26] NAU, D.; GHALLAB, M.; AND TRAVERSO, P. 2004. *Automated Planning: Theory & Practice*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- [27] PRATT, G., AND WILLIAMSON, M. 1995. Series elastic actuators. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, 399–406.
- [28] SHAW, M. 2002. What makes good research in software engineering? In *Proceedings of the European Joint Conference on Theory and Practice of Software*.
- [29] SIMMONS, R.; GOLDBERG, D.; GOODE, A.; MONTEMERLO, M.; ROY, N.; SELLNER, B.; URMSON, C.; SCHULTZ, A.; ABRAMSON, M.; ADAMS, W.; ATRASH, A.; BUGAJSKA, M.; COBLENZ, M.; MACMAHON, M.; PERZANOWSKI, D.; HORSWILL, I.; ZUBEK, R.; KORTENKAMP, D.; WOLFE, B.; MILAM, T.; AND MAXWELL, B. 2003. Grace : An autonomous robot for the AAI robot challenge. *AI Magazine* 24(2):51–72.
- [30] SMART, W. D.; DIXON, M.; MELCHIOR, N.; TUCEK, J.; AND SRINIVAS, A. 2003. Lewis the graduate student: An entry in the AAI robot challenge. Technical report, AAI Workshop on Mobile Robot Competition. p. 46–51.
- [31] STIEHL, W., AND BREAZEAL, C. 2005. Design of a therapeutic robotic companion for relational, affective touch. In *Proceedings of the IEEE Workshop on Robot and Human Interactive Communication*.

- [32] VALIN, J.-M.; MICHAUD, F.; AND ROUAT, J. 2006. Robust localization and tracking of simultaneous moving sound sources using beamforming and particle filtering. *Robotics and Autonomous Systems Journal* 55:216-228.
- [33] VALIN, J.-M.; ROUAT, J.; AND MICHAUD, F. 2004. Enhanced robot audition based on microphone array source separation with post-filter. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2123-2128.
- [34] VAUGHAN, R. T.; GERKEY, B. P.; AND HOWARD, A. 2003. On device abstractions for portable, reusable robot code. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2421-2427.
- [35] WERNER, M. 1999. *Humanism and beyond the truth*, volume 13. Humanism Today. <http://www.humanismtoday.org/vol13/werner.html>.

Received: November, 2007

Accepted: April, 2008

Contact addresses:

François Michaud
 Dominic Létourneau
 Maxime Fréchet
 Michel Lauria
 Department of Electrical Engineering
 and Computer Engineering
 Université de Sherbrooke
 Québec, Canada
 e-mail: francois.michaud@usherbrooke.ca

Éric Beaudry
 Froduald Kabanza
 Department of Computer Science
 Université de Sherbrooke
 Québec, Canada

FRANÇOIS MICHAUD is a Professor at the Department of Electrical Engineering and Computer Engineering of the Université de Sherbrooke, Québec, Canada. He is the Canada Research Chairholder in Mobile Robots and Autonomous Intelligent Systems. His research interests are in architectural methodologies for intelligent decision-making, design of autonomous mobile robotics, social robotics, robot for children with autism, robot learning and intelligent systems.

DOMINIC LÉTOURNEAU is a Research Engineer at the Department of Electrical Engineering and Computer Engineering of the Université de Sherbrooke, Québec, Canada. His research interests cover combination of systems and intelligent capabilities to increase the usability of autonomous mobile robots in the real world. His expertise lies in artificial vision, mobile robotics, robot programming and integrated design.

ÉRIC BEAUDRY is a Ph.D. student at the Department of Computer Science of the Université de Sherbrooke, Québec, Canada. His expertise is in artificial intelligence, planning and scheduling and autonomous robotics.

MAXIME FRÉCHETTE is a Master's student at the Department of Electrical Engineering and Computer Engineering of the Université de Sherbrooke, Québec, Canada. His expertise lies in dialogue management and human-robot interaction.

FRODUALD KABANZA is a Professor at the Department of Computer Science at the Université de Sherbrooke. His research interests include intelligent decision-making, AI and planning, AI and education and user-modelling.

MICHEL LAURIA is a Professor at the Department of Electrical Engineering and Computer Engineering of the Université de Sherbrooke, Québec, Canada. His research interests are in mobile robot design and control, mechatronic and high-performance actuators.
