Journal of Intelligent and Robotic Systems (2005)
DOI: 10.1007/s10846-005-9018-0

Intelligent Human–Machine Interaction Based on Dynamic Bayesian Networks Probabilistic Intention Recognition

KARIM A. TAHBOUB
Palestine Polytechnic University, Hebron, Palestine; e-mail: tahboub@informatik.uni-freiburg.de

(Received: 12 November 2004; in final form: 2 September 2005)

Abstract. In this article, a novel human–machine interaction based on the machine intention recognition of the human is presented. This work is motivated by the desire that intelligent machines as robots imitate human–human interaction, that is to minimize the need for classical direct human–machine interface and communication. A philosophical and technical background for intention recognition is discussed. Here, the intention–action–state scenario is modified and modeled by Dynamic Bayesian Networks to facilitate for probabilistic intention inference. The recognized intention, then, drives the interactive behavior of the machine such that it complies with the human intention in light of the real state of the world. An illustrative example of a human commanding a mobile robot remotely is given and discussed in details.

Key words: compliant interaction, dynamic Bayesian networks, human–machine interaction, human–robot interaction, intention recognition.

Categories: intention recognition, intelligent systems, man–machine interaction, teleoperation.

1. Introduction

In their final report for the DARPA/NSF interdisciplinary study on human–robot interaction [5], Burke and her colleagues identified grand challenges facing sought intelligent human–robot interaction. Intention recognition was identified to be among the first group of technology challenges: “Picking up human social cues (attentional stance, body language) and interpreting human behavior (intent, goals, desires) would be impressive demonstrations.”

In many multi-agent systems, when two agents or more cooperate or compete to achieve a certain task, the problem of recognizing the intention of others arises. Usually, the lack to a reliable and full communication of intentions calls for inferring the intention from the actions or even from the changes in the environment due to agent actions.

Consider the example of a human operator sitting on and commanding a robotic wheelchair with extended perceptual and actuating capabilities. The robotic wheelchair can navigate autonomously in the environment. However, because of the interaction with the human, a semi-autonomous system results; the wheelchair senses the environment and tries to comply with the human commands while protecting her from any collision with external bodies. In other words, the wheelchair gives up a part of its autonomy in favor of better cooperation with the human user. However, without recognizing the real intentions of the human, the wheelchair can behave in a less cooperating and a more frustrating way. On the other hand, it is impractical and annoying to ask the human operator to communicate her intentions explicitly either by speech or through other interfacing devices. Thus, it is favorable that the wheelchair itself infers the actions (commands or their effects) of the human into intentions [22]. There are many other applications that call for intelligent intention recognition to improve the human–machine interaction including intelligent cars and intelligent transportation systems, physical training machines, disabled-people-assistive technologies, and home appliances.

In the literature, intention recognition and plan recognition are used interchangeably while meaning the same thing. However, in this work, a distinction is made between intentions and plans. Whilst philosophers like Bratman [2] consider intention as the main attitude that directs future planning, intentions here will be considered to direct specific actions. Accordingly, a ‘plan’ is composed of a sequence of conditional actions to achieve a certain goal whereas an ‘intention’ is the explanation behind a specific action. In other words an intention can be thought of as a ‘label’ to a certain action. Actions themselves can be simple atomic or complex.

1.1. LITERATURE REVIEW

Initial work on plan recognition grew out of work on the problem of understanding natural language narrative [23]. The first attempt to find a generalized framework was in the generalized plan recognition paper of Kautz and Allan [11]. Techniques from inductive concept learning were applied to plan recognition by Lesh and Edzioni [12].

Charniak and Goldman argued that the problem of plan recognition is largely a problem of inference under conditions of uncertainty [6]. Although the main advantage is to reduce the number of top-level plans, this view forms a solid ground for physical applications where uncertainty is inherent in both the actuation and perception. Later Pynadath [19] applied Bayesian nets approach to the traffic monitoring problem.

Recently Bui et al. [4] introduced the abstract hidden Markov model to obtain an online probabilistic algorithm for plan recognition. On the other hand Mulder and Voorbraak [15] extended previous works on classical abduction to be applied to plan recognition.

Plan recognition has been applied to different domains. Among these are story understanding [6], automated driving and traffic monitoring [19], human–computer interface [13], military [21], and aviation monitoring [8].

1.2. INTENTION RECOGNITION OVERVIEW

Several philosophers like Bratman [2, 3] and Dennett [7] discussed the role of intention in theories of consciousness, planned action, rationality, and intelligence. As mentioned, Bratman considers intention as the main attitude that directs future planning. Dennett on the other hand argues that the ascription of intent is a perfectly reasonable way of predicting and describing the behavior of systems that are complex enough to avoid explanation from other stances [8]. He offers three stances that can be used to explain and predict the behavior of the system: the physical stance that operates directly upon knowledge of physical composition and of the laws of physics, the design stance that deals with the purpose the system is designed for, and the intentional stance where a system is assumed to have certain beliefs and desires and it behaves in such a way to further its goals in light of its beliefs.

Human interaction usually requires continuous and complex intention recognition. In simple conversations, for example, humans try to predict the future direction of the conversation and the reaction of the other person by recognizing the intent manifested in the conversation. Further, in a promenade, two persons mutually coordinate their steps and moves by predicting the intention of each other. Often intention recognition by humans is performed subliminally; conversations unfold and subtle queues are reacted to without much conscious thought given to the underlying motives of the other [8].

Intention recognition can be seen as a substitute or complimentary to reliable and extensive communication which is a prerequisite for coordination and cooperation. If agents are able to express their intent clearly and honestly then intention recognition reduces to communication. But since not all agents are explicitly aware of their intentions or since communication can be a burden on agents (different designs, different levels of intelligence, or heterogeneous ontologies) then intention recognition becomes essential. Furthermore, it is sought to have a natural interaction in human–machine cooperation; natural to a level that resembles human–human interaction.

Intention recognition is defined, in general terms, as the process of becoming aware of the intention of another agent. More technically, it can be defined as the problem of inferring an agent's intention through its actions and their effects in the environment. It lies accordingly in the boundary between perception and cognition.

It is generally not required that the agent whose intention is being recognized has any explicit internal representation of intention or even an explicit intent.

This is true as intention can be ascribed. However, agent's rationality is usually a common assumption in intention recognition. It is claimed in this article that rationality itself does not necessarily imply consistency in behavior and action. This will be illustrated in upcoming sections.

Heinze [8] proposed a tri-level decompositional description of intentional behavior of the intending agent. Intention recognition is the reversal of this process. These levels are:

- The intentional level that describes the intention in terms of beliefs, goals, plans, desires, etc.
- The activity level that describes the task, function and actions.
- The state level that describes the agent in terms of externally accessible characteristics and effects its actions have in the environment.

According to this description, six basic paths (approaches) can be followed in intention recognition as shown in Figure 1. For example, scheme 1 corresponds to the basic 'sense and infer' approach, scheme 3 matches the trivial case of communicating intentions, while scheme 6 resembles direct action recognition from state and inferring the recognized action in intention.

This paper describes a modified approach similar to scheme 1 where inference is done through Dynamic Bayesian Networks. In this work, a fourth level corresponding to the desired state is added. Thus the intentional level is decomposed into two levels: intention and desired state. The desired state, either explicitly or implicitly represented, is the final result of the cognitive and planning process and is seen as its output. The actions, therefore, are selected to achieve the desired states. This decomposition is more modular as will be seen in the sequel. Figure 2 shows the adopted intention recognition scheme.

2. Dynamic Bayesian Networks

Probabilistic reasoning approaches have profound advantages in application domains where uncertainty is intrinsic. Agents usually act under uncertainty as they can never have a full and accurate perception of their environments. Furthermore, rational agents do not usually further their goals in a simple if-then fashion.

Bayesian reasoning in general and Bayesian artificial intelligence in particular are based on Bayes' theorem:

$$P(h|e) = \frac{P(e|h)P(h)}{P(e)} \tag{1}$$

It asserts that the probability of a hypothesis h conditioned upon some evidence e is equal to its likelihood $P(e|h)$ times its probability prior to any

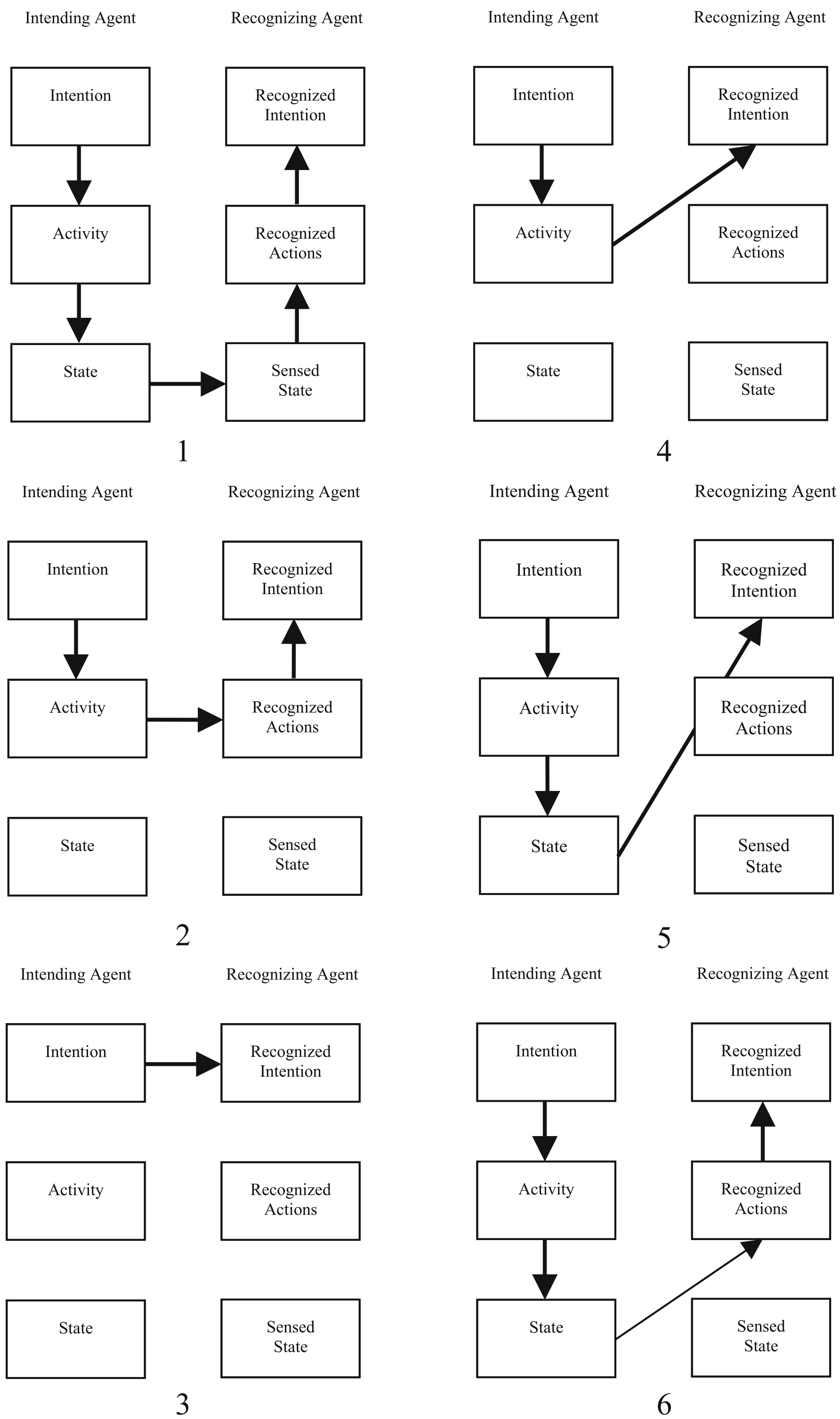

Figure 1. Six possible paths in intention recognition [8].

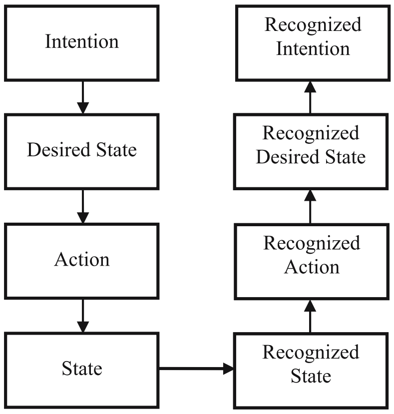

Figure 2. Extended intention recognition scheme.

evidence $P(h)$ normalized by dividing by $P(e)$ (so that the conditional probabilities of all hypothesis sum to 1). In other words and in a causal setup, it is possible to find the posterior probability of a certain cause given the likelihood of its effect. Hence, Bayes' theorem forms an excellent basis for inferring the intention from the sensed states. However, in most practical cases, the relationship between the intention and the state is not a direct and simple one. Accordingly, it becomes essential to use a representation that admits many interconnected variables and stages and makes use of conditional independence of variables. Two variables x and y are said to be conditionally independent if and only if

$$\begin{aligned} P(x|y) &= P(x) \\ P(y|x) &= P(y) \end{aligned} \tag{2}$$

A suitable representation is Bayesian network. A Bayesian Network (BN) is a directed acyclic graph encoding assumptions of conditional independence. Nodes in BN represent stochastic variables whereas arcs represent causal dependence. Associated with each node is a specification of the distribution of its variable conditioned on its predecessors in the graph. Such a network defines a joint (conditional) probability distribution (CPD); that is the probability of an assignment to the stochastic variables is given by the product of the probabilities of each node conditioned on the value of its predecessors according to the assignment. For example, for a simple network composed of three nodes x_1, x_2, and x_3 where x_1 and x_2 are the parents of x_3, the probability becomes:

$$P(x_1, x_2, x_3) = P(x_1)^*P(x_2)^*P(x_3|x_1, x_2). \tag{3}$$

So, to define a BN, one needs to specify the structure of the network, to specify the conditional probability distribution and finally to specify the prior probability distribution of the top nodes.

In using BNs for intention recognition, intentions are represented by top nodes. The result of the intentional deliberations is a desired state. Actions and changes in states that follow from these intentions and desired states are represented by nodes below connected causally to intention nodes. Observable effects (states as measured by sensors) occupy usually bottom nodes. The prior probability distribution of the intention nodes reflects the problem context or the intending agent mental state.

Domain experts can decide on the structure of the network (which variables have which effect on other variables) and can specify the required CPD. It is however possible to learn the structure and the accompanying CPD automatically. Intentions can, thus, be recognized by a bottom-up inference. Here, only the values of the observed nodes are used to compute the posterior probability distribution for intention nodes.

To take into account the temporal dynamics of the 'intention–activity–state' process, the Dynamic Bayesian Network (DBN) is used. DBN is a special BN architecture for representing the evolution of variables over time. It consists of a sequence of time slices where each time slice contains a set of variables representing the state at the current time. A time slice is in itself a BN, with the same network structure replicated at each time slice. The temporal dynamics of the process is encoded via the network links from one time slice to the next [4]. To construct a DBN, one must specify three kinds of information: the prior distribution over the variables, the transition model (from one time slice to the next), and the conditional probability distribution.

3. Human's Intention Recognition

Figure 3 shows a feedback schematic diagram representing the four levels. The intention level depends on the intention node of the previous time slice (dashed arrow) together with the current world state. The action, on the other hand, is the

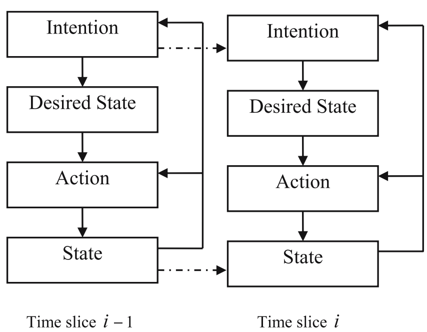

Figure 3. Human's intention–action–state flow.

causal outcome of the desired state and the current world state. This is analogous to a classical control-system feedback loop. Finally, the current state is obtained by the performed action taking into consideration the previous state; this is analogous to integrating a governing differential equation given its initial conditions and external input. The lower parts of the diagram are modular in the sense that they can be automated.

As human intentions and actions depend on the state of the environment, cycles appear in the network. However, as a DBN is acyclic graph by definition, one cannot add cycles without expecting difficulties. Actually, all known exact inference algorithms assume 'acyclicity'.

To go around the problem, different solutions were proposed in the literature:

1. Converting the graph with cycles to a junction tree and using exact belief propagation methods. This is possible for a class of graphs with certain conditions [16].
2. Applying belief propagation to the graph with cycles and obtaining an approximate inference. This is first suggested by Pearl in 1988. For details, see the work of Yedidia et al. [24].
3. Recently, Peng and Ding [18] proposed an approximate approach to eliminate cycles using minimal likelihood loss (a short cycle first heuristic).

In this work, a novel method of eliminating cycles by time delay is proposed. As the cycles arise due to the feedback of sensed states to the intention and action nodes, one can eliminate the cycles by feeding back the sensed states from a previous time slice instead of the current one. This is quite logical as the sensed states are not instantaneously attainable but become available only after some time delay. This trick can be applied to many cases when DBN is used. Figure 4 shows the modified version of the original schema as depicted in Figure 3. This way, the cycles are removed from the DBN without any approximation or transformation. Even in cases when cycles are instantaneous, this approach serves as an excellent approximation.

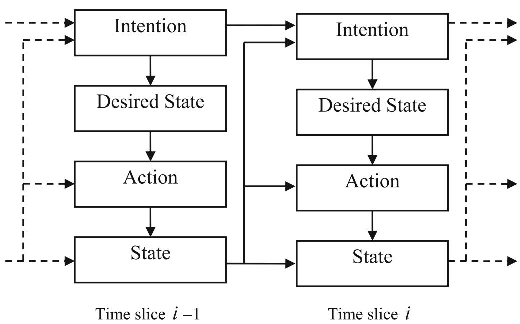

Figure 4. DBN corresponding to human model with time-delay (no cycles).

4. Compliant Human–Machine Interaction

There has been a lot of methodologies presented in the literature dealing with the issues of human–machine interaction (HMI), human–computer interaction (HCI), and human–robot interaction (HRI). It is not the scope of this current work to review these methodologies neither to establish new foundations for this important field. The ultimate goal of this work is to incorporate the intention recognition module into human–machine interaction to achieve compliant and intelligent interaction. For this, an architecture for compliant human–machine interaction is proposed in Figure 5. Classical modules as human–machine interface and supervision and control can be integrated in this architecture. Although one of the goals of intention recognition is to minimize the need for traditional interfaces, it is still possible to augment certain interaction aspects with visual and auditory interfaces. General aspects of human–machine interfaces are discussed in [10]. On the other hand, human supervisory control can be seen as a general framework for delegating originally human tasks to the machine. Although the intention recognition module lies roughly in this category, it is distinguished as a separate module as it relies on a different taxonomy than that detailed by Sheridan [20]. Inagaki and Stahre [9] give an interesting overview of supervisory control.

In the proposed architecture, the processes in the world are divided to human processes, machine processes, and shared processes. The effect of operating these processes can be partially or completely sensed by the human and the machine as world states. The human uses the sensed information to further her goals by selecting actions to achieve her desired states. On the other hand, the machine which has a human-task model, employs the sensed states to recognize the human intention and accordingly to select its action in a compliant fashion with

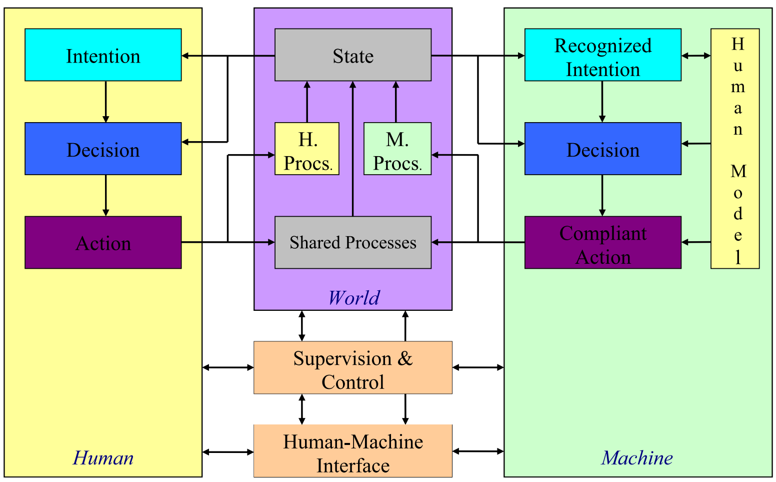

Figure 5. Compliant human–machine interaction architecture.

the human intentions. The previously mentioned division of world processes gives a great flexibility in the design as it incorporates manual mode, autonomous mode, and semi-autonomous (shared-control) mode. This flexibility widens the architecture application domain. However, and as the machine actions can be assumed to be known for the machine, the intention inference problem remains solvable despite this great flexibility as discussed in the coming section. It is notable here that the relation between the machine and the world is fuzzy. The processes can be physically a part of the machine and the state can be partially the machine state, nevertheless the distinction between the machine and the world is made to highlight the machine agency and autonomy.

Even in the case of shared processes, it is possible that human and the machine have the same action channels, i.e., the actual action is the summation of both actions. Depending on the application, the summation can be a simple algebraic addition or a complex operation in the action space. However, if each of them can change the world state through a dedicated action channel then the summation should come after the individual sate level. In general, the human can change the world state either directly or by commanding the machine. For example, when a human is mounting a car door with the help of a robot, the human can physically carry the door and thus changes its state or command the robot by a teach pendant to change its state. To avoid any confusion, one can think of the machine as playing two roles at the same time: fully autonomous by generating its own actions (even when driven by recognized intentions) and fully manual in translating human actions (commands) into physical state changes.

5. Implementation

Before presenting an illustrative application example, some implementation issues are presented:

A. *Human Action Model*

This block can contain a mental model, perception model, and a motor model of the human. The issue of varying human behavior can be addressed in this part as well. It suffices, however for intention recognition purposes, to ideally assume rationality and perfect perceptual and motor capabilities. Any deviation from the ideal model can be observed and compensated by the intelligent machine. This perfection assumption can be alleviated in future works by adapting this model to different human users either through learning or incorporating a proficiency factor.

B. *Compliant Machine Action Model*

The machine complies with the recognized intention by adjusting its actions in accordance with the sensed world state. If the human action is partially (temporally or in the state space) inconsistent or not fully efficient to achieve the recognized intention, then the machine overcomes this action by applying an extra action. This behavior is the source of intelligent and compliant behavior. It is compliant because it is driven by the human intentions and it is intelligent because it takes into account the real state of the world.

There is a need for an arbitration mechanism to perform the selected human and machine actions. Subsumption or motor-schema similar methods can be applied depending on the application. This does not contradict with the spirit of the proposed architecture as the machine actions are selected always in accordance and compliance with the human intentions.

C. *Inference while Actively Interacting with the Human*

The problem of inference for human intentions when the machine just ‘observes’ is a relatively simple problem. However, it is expected that the machine interacts with the human by performing certain actions and thus changes the state of the world. Accordingly, as a reaction, the human may change his actions, desired states or even intention. This leads to a more involved inference problem. In fact, since it is realistic to assume that the machine knows its actions (as well as recognized intentions and desired states) in a deterministic fashion, the inference problem becomes bigger only in terms of the number of added nodes while the quality of inference (observability) remains unchanged as the added nodes are observable. It may be feasible to condition the human intention on the machine action by adding a causal link from the machine action nodes to the human intention node.

6. Application Example

While it seems effortless for humans to perceive spatial objects and reason about their relations, this is still difficult for computers. Part of the computational problem lies in the difficulty of identifying and manipulating qualitative spatial representations [1]. Handling spatial data is a key task in many applications including mobile robots, teleoperation, and aviation systems. Intention recognition gains more importance and becomes more difficult in spatial interaction. Qualitative meaningful intentions are to be selected and recognized in order to facilitate the human–machine interaction. Consequently, an example related to spatial human–robot interaction is presented. Two cases can be thought of: human sitting on and commanding a robotic wheelchair and a human operating a mobile robot remotely.

Consider a human trying to manipulate a mobile robot in an environment. The robot is assumed holonomic and equipped with a laser scanner, sonars, and a camera. The human, who has access to a not-delayed robot-environment image, manipulates the robot remotely by applying forces on it through a joystick. Here, the dynamics of the robot, excited by those forces, dominates its response as it has a mass and moves on a surface with friction. In consequence, the human must be well trained to be able to manipulate it without colliding with objects or taking much time to achieve the goal. Three human limitations subdue the operation: far from perfect accelerating and braking maneuvers, delayed human response, and inconsistent (fluctuating, spatially confused, ...) commands. It becomes then clearly advantageous that the robot recognizes the intention of the human and complies with it.

In order to simplify the discussion and the presentation, it is assumed that the space is composed of objects (existent and virtual). The motion of the robot is then described in relation to those objects. Four atomic intentions are considered: move to an object, move parallel to an object, move away from an object, and do nothing regarding an object. These intentions are atomic as they do not constitute plans and they do not require full knowledge about the environment; local sensor information suffices, namely distance and direction to object and robot velocity relative to the object. The human input (deriving the robot) is dissolved into two components (in direction of an object and orthogonal to it). Figure 6 shows the BN (only one time slice) corresponding to this situation. It consists of nine nodes corresponding to the four proposed levels:

1. Intention: intention node I to denote the four discrete intentions (G: move to an object, R: move away from an object, P: move parallel to an object, and N: do nothing regarding an object).
2. Desired state: desired velocity in the x and y directions relative to the object (D_{Vx} and D_{Vy}) and desired distance to the object D_D. The x direction is selected as the shortest line to the object.
3. Action: applied forces in the x and y directions (F_x and F_y).
4. State: sensed velocities (V_x and V_y) and sensed distance D.

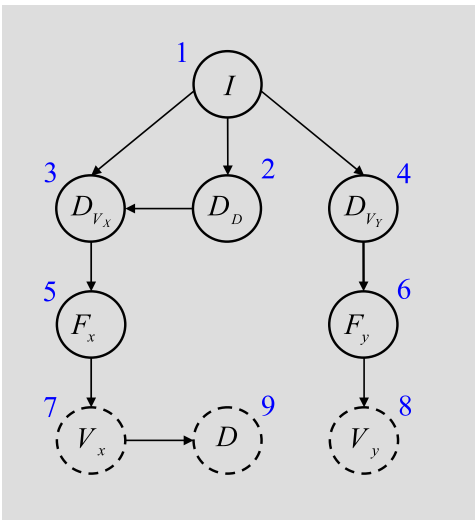

Figure 6. BN corresponding to human commanding a robot.

The arrows in the network correspond to causal relationships. D_{Vy}, for example, is set according to the intention while it, at the same time, results in applying a force F_y. This last force causes the robot to move with velocity V_y. The two directions x and y are chosen in such a way to decouple the forces. All network stochastic variables (nodes) are chosen to be discrete to simplify the inference as discussed later. The causal relationships are characterized by quantitative probabilistic models in the form of, for example:

$$P(D_{V_x} = \text{low\_positive}/I = G\&D_D = \text{medium}) = 0.7\,. \tag{4}$$

These relationships are gathered finally in Conditional Probability Distribution (CPD) tables. The topology of the network together with the CPDs defines the network. The network topology and its CPDs can be learned automatically; nonetheless it is quite easy to set them manually by common sense for this abridged example.

The observed nodes are denoted by dashed circles while the hidden nodes are denoted with solid ones. The hidden nodes, especially the intention node, are inferred given the value of the observed nodes.

Since this is essentially a dynamic process, temporal dimension should be added to the network. Figure 7 shows the corresponding DBN (two time slices; the first is the BN while the second is shown in the figure). The first time slice is connected to the second shown in the figure by the dashed lines. Note that no cycles exist as feedback from previous time slice is used instead of the current time slice. Extra CPDs are given for nodes changed in the second time slice as

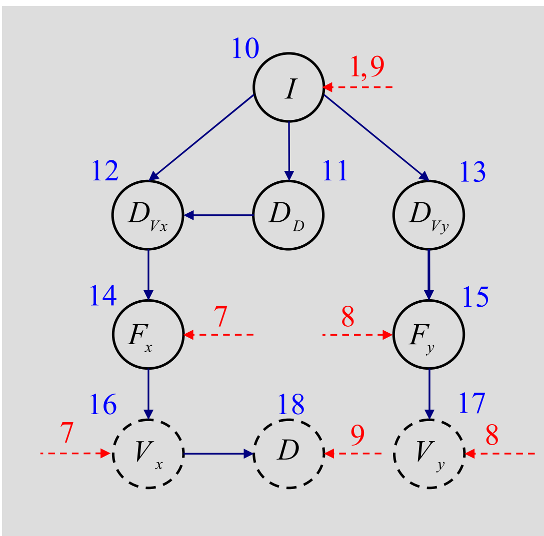

Figure 7. DBN corresponding to human commanding a robot.

Table I. Conditional probability distribution of node 14.

D_Vx '12'	Vx '7'	P(Fx = − −)	P(Fx = −)	P(Fx = 0)	P(Fx = +)	P(Fx = ++)
− −	− −	30	70	0	0	0
− −	−	50	50	0	0	0
− −	0	70	30	0	0	0
− −	+	90	10	0	0	0
− −	++	100	0	0	0	0
−	− −	0	20	80	0	0
−	−	20	80	0	0	0
−	0	50	50	0	0	0
−	+	70	30	0	0	0
−	++	90	10	0	0	0
0	− −	0	0	0	80	20
0	−	0	0	50	50	0
0	0	0	0	100	0	0
0	+	0	50	50	0	0
0	++	20	80	0	0	0
+	− −	0	0	0	10	90
+	−	0	0	0	30	70
+	0	0	0	0	50	50
+	+	0	0	0	80	20
+	++	0	0	80	20	0
++	− −	0	0	0	0	100
++	−	0	0	0	10	90
++	0	0	0	0	30	70
++	+	0	0	0	50	50
++	++	0	0	0	70	30

for example nodes 10, 14, and 15. As illustration, Table I shows the CPD corresponding to node F_x in Figure 7 (or equivalently node HF_x in Figure 9) which gives the probability distribution of the applied force given the desired velocity and the actual velocity. The numbers listed in the table are selected manually according to the understanding of the behavior. There, '− −' denotes large negative, '−' denotes small negative, '0' denotes around zero, '+' denotes small positive, and '++' denotes large positive. This probability distribution models the rational behavior of a human commanding the machine by applying a force given her desired velocity and the actual velocity. This resembles a human classical feedback control problem.

So far, the human part of the architecture is discussed. Now, to integrate the machine part, extended BN and DBN are shown in Figures 8 and 9. The robot action is denoted by M, the human action by H, and the total action by T. There, only the last level of the machine part (the action) is included. This is justified as it is assumed that the robot action is known to it, i.e. the two nodes MF_x and MF_y are observable. These nodes constitute the final result of robot decision cycle and form the interface to the human part. Hence, the machine nodes, which lie above

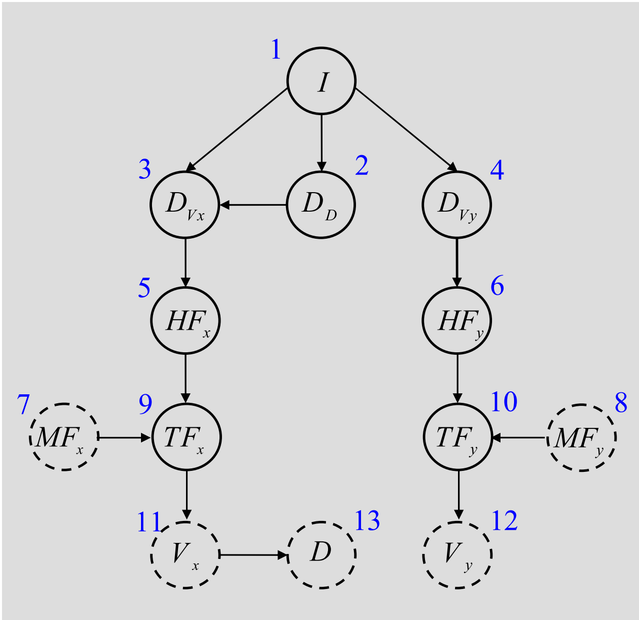

Figure 8. BN corresponding to human–robot interaction.

the action nodes, are of no interest to the inference problem. The rules governing the above nodes are given in the sequel. The networks assume that the process is a shared one where the human and robot actions can be summed. Here, the motor-schema methodology is adopted where the total action is taken to be the

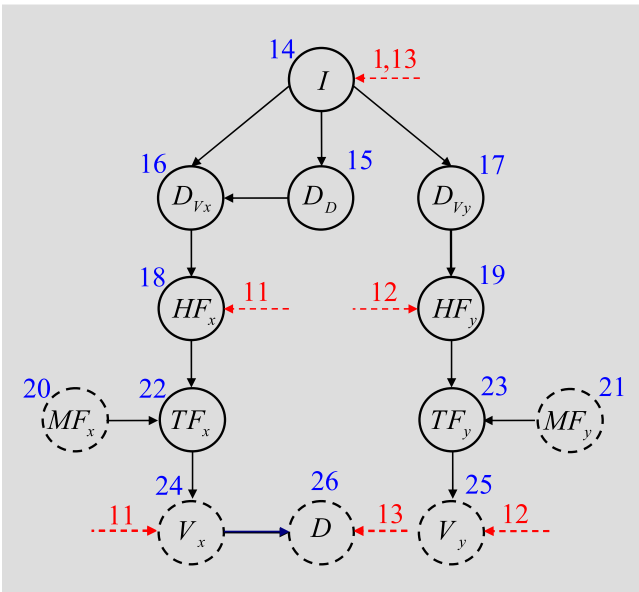

Figure 9. DBN corresponding to human–robot interaction.

average of the human and robot actions. Nevertheless, it is possible to adopt any suitable arbitration method (subsumption or semi-autonomous architectures). The method adopted is then to be modeled by fixing the equivalent conditional probability distribution for TF_x and TF_y.

For this specific example, the robot has a mass of 30 kg and a damping ratio of 15 kg/s. Furthermore, the maximum applied force is limited to 30 N. Consequently, the following robot derivative control loop generates reasonable response:

$$MF_x = 15*(MD_{V_x} - V_x) + 15*MD_{V_x} \tag{5}$$

where MF_x is the robot-applied force, MD_{Vx} is the robot-desired velocity, and V_x is the measured velocity. Force saturation is cascaded with the control law to keep the force within its maximum. The same holds for the y direction. The robot desired velocities, which are needed for equation (5), are selected in compliance with the recognized intention and based on the sensed velocity and distance:

$$MD_{V_x} = \begin{cases} 0.2*D & I = G \\ -2/D & I = R \\ 0 & I = P \text{ or } I = N \end{cases} \tag{6}$$

$$MD_{V_y} = \begin{cases} 0 & I = G \text{ or } I = R \\ V_y & I = P \text{ or } I = N \end{cases} \tag{7}$$

This simple compliant robot behavior compensates for imperfect human action. For example, it is responsible for reducing the speed when approaching an object by applying a negative force depending on the actual measured speed. If more than one object is identified, the total robot force is selected as the average of all forces corresponding to those objects. This summation is the source of an emerging intelligent behavior. The robot selects its path between objects in a non-expected smooth way.

To test the validity of the proposed methods, an animation environment is built in MATLAB using its graphical user interface utility. A robot workspace with variable objects (labeled 1 to 4) and walls with doors and corridors is setup as shown in Figure 10. The robot is given sensing capabilities of objects close to it (rays going out of the robot correspond to sonar rays) and can sense its own speed. A human commands the robot by applying forces on it by moving a joystick (this joystick is shown as a circle in upper right of the figure with the position of the joystick in second quarter). The machine action is visualized by an automatically moving joystick (this joystick is shown as a circle under the first joystick, at the instance shown the machine is not acting as the joystick position is in the center). The inference algorithm accesses information only about the observable nodes. It employs a discretized version of these stochastic variables based on predefined discretization rules. For the inference, a smoothing engine

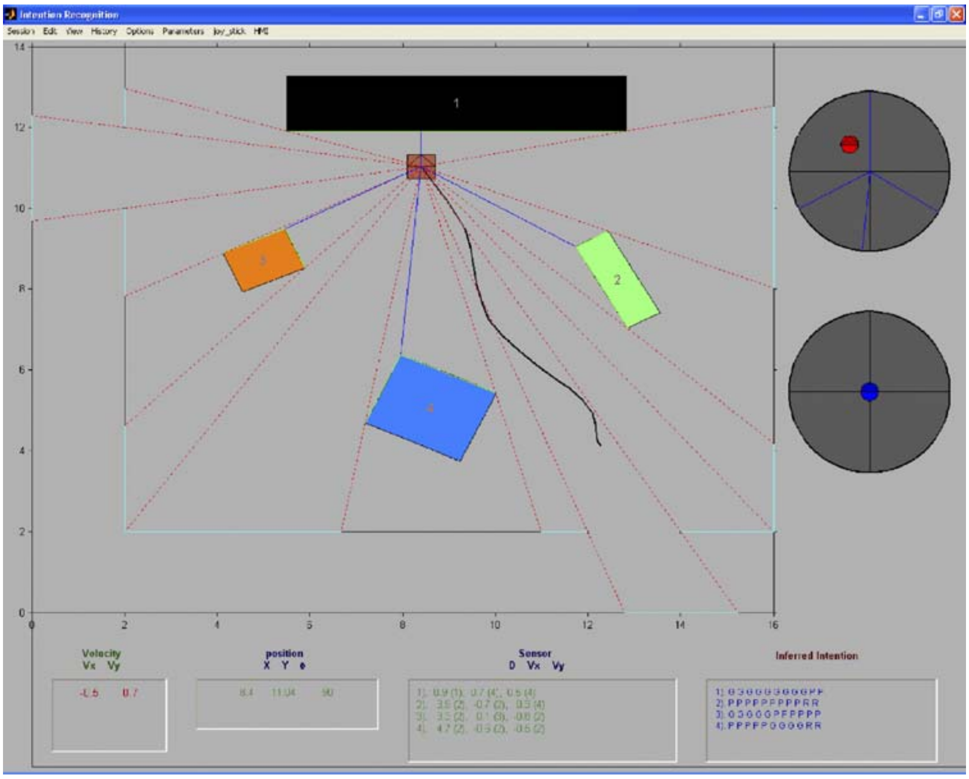

Figure 10. Mobile robot animation environment.

based on a junction tree inference engine is used. This is supported in the MATLAB toolbox 'BNT' developed by Murphy [17]. The absolute robot state (position and velocity) together with that relative to the different objects is displayed on the screen. The motion intention regarding each sensed object is displayed as well. The joystick driven by the human is shown, below which is shown, as well, a joystick corresponding to the robot action. The average of the two is used to drive the robot. These forces are substituted in the dynamics equations to calculate the robot velocity and position.

Different setups are experimented. The intention recognition converges always to the right values. However, depending on the chosen inference update rate and the number of previous samples fed to the inference function (environment parameters); the intention recognition module can sometimes generate inconsistent transient results. For example, if the update rate is very high and the number of samples used for inference is small (2 or 3), then the inference becomes hyper sensitive with no memory and thus can generate results corresponding to instantaneous small changes in the observed variables. However, this is improved by using more samples and choosing an update rate appropriate to the system time constant.

It is difficult to describe by words and graphs the behavior of the system without actually having access to the animation environment. Nevertheless, a simple example is discussed in details to highlight the actual human–robot cooperation. The robot is located at the bottom left corner of the space and a long object covers the upper wall of the space shown in Figure 10 (so, there is only one object in the space far away from the robot). The task is to move the robot toward the object (I = G), then to move it parallel to the object in the right direction (I = P, V_y is negative), then to move it parallel to the object in the opposite direction (I = P, V_y is positive), then finally to move it away from object to its initial location (I = R). On purpose, the human does not command the robot perfectly. Figures 11–13 show data obtained from the GUI environment. The recognized intention as shown in Figure 11 corresponds well with the actual intention. It starts with N for one sampling time, followed by G, then by P, and finally by R. The human applied force in the direction of the object is shown together with the robot's force in Figure 12. The perfect human behavior would be to accelerate at the beginning and then to decelerate in order to reach the object in a short time without colliding with the object. It is clear from the simulation that the robot complies with the human intention as it exerts the ideal force according to equation (5). In the second stage, the robot tries to keep the

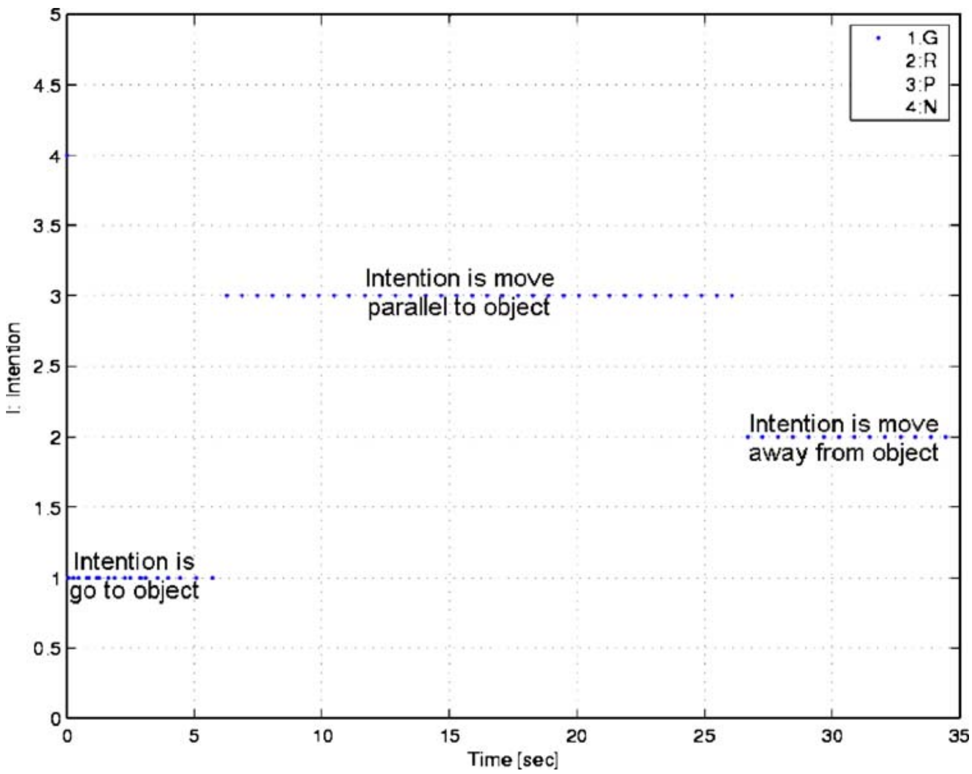

Figure 11. Recognized intention.

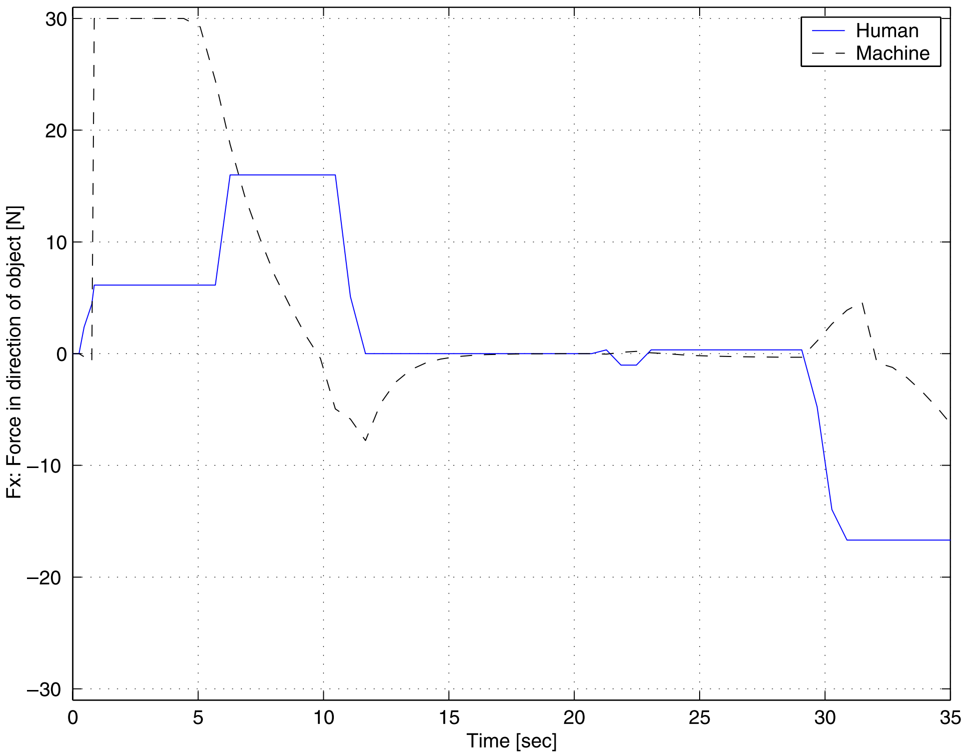

Figure 12. Human and machine applied forces in the direction of object.

distance to the object constant; this can be seen in Figure 12 with the positive force (MF_x) to compensate for the negative force applied by the human. In the stages (G, R), the ideal thing is to minimize the forces perpendicular to the object; the effort of the robot to do this can be noticed in Figure 13. For the (P) stage, the robot tries to exert the same force as the human does; it is desired that only the human decides on the speed of motion parallel to the object. However, the robot has access only to the observable states without knowing the applied human action. So, it senses the actual velocity (V_y) and applies a force to maintain this velocity according to equations (5) and (7). This effort can be observed as well in Figure 13. The distance trajectory is a relatively smooth one despite the sudden change in the human's command. The velocity profiles are close to the desired ones set by equations (6) and (7).

For all parts of this work, discrete stochastic variables (nodes) are assumed. This has the advantage of testing and using different available inference engines (methods) and the disadvantage of losing valuable information. In principle, the intention inference algorithm depends on observing the applied robot action to "eliminate" its effect in the process of recognizing the human intention through her actions. However, the assumed fully available information about the applied robot action is available only in a discrete coarse form. This leads to a loss of

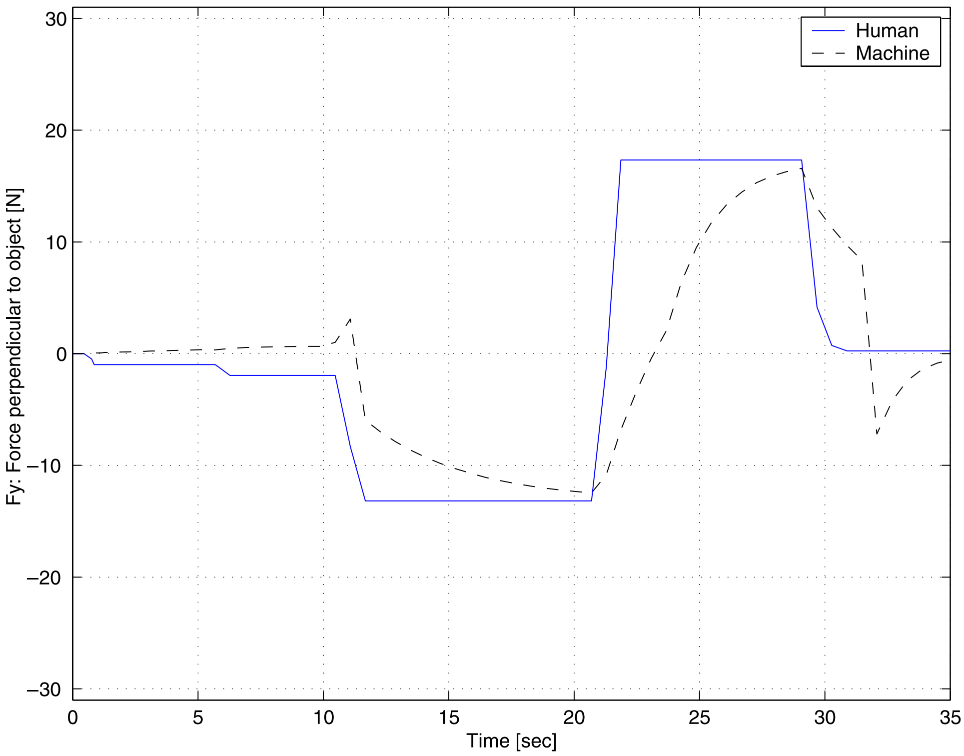

Figure 13. Human and machine applied forces perpendicular to object.

valuable information which in some cases misleads the inference process. That is the inference algorithm does not know about varying (fluctuating) robot action that lies within the same discrete level. As a result, it is advantageous in terms of accuracy either to increase the number of discrete levels (at the cost of bigger CPD tables) or to employ continuous representation.

7. Conclusions and Future Work

In this article, the novel idea of compliant human–machine interaction based on human intention recognition is introduced. Intention recognition and human–machine interaction modalities are introduced to achieve the goal of human–machine interaction similar to human–human interaction.

The term ‘intention recognition’ is explained; philosophically related issues are discussed, moreover, the corresponding literature review is given. Here, the classical intentional level is decomposed to intention and desired state levels. This makes the human deliberation in the environment analogous to a feedback system where she tries to minimize the difference between the desired and actual sensed states. The problem of arising loops (due to feedback) in the DBN is solved by a novel method of introducing a time delay to the network.

To avoid the difficulty of finding suitable CPDs, especially when the size of the DBN is big and its structure is complex, it is possible to learn the structure and its parameters. Most available Bayesian network software packages provide learning algorithms. The proposed architecture in this article is a general framework; for a specific domain, nodes corresponding to each level must be determined and causal relations must be set. The intentions themselves can be higher level goals as ‘go to another room.’ In that case, actions are more abstract and can be divided to a series of atomic actions.

A human–machine interaction architecture that enables for basing the machine action on the recognized human intention together with the state of the world is presented. The processes in this architecture can be fully autonomous, fully manual, and shared-control processes. The architecture enables, as well, both the human and the machine to change the state of the world either directly or through common action channels. The machine acts in a compliant fashion with the human intentions according to a predefined ‘ideal’ action model while taking into account the state of the world.

An illustrative example of navigating a mobile robot is given. For that, a MATLAB animation environment is built. The environment enables for interactively adding and moving arbitrary objects, collision avoidance, intention recognition, and human–machine interaction. The user can activate/deactivate the machine interaction which leads to calling the corresponding intention recognition algorithm. The intention recognition performs well and reliably even in the case of the machine interaction. The presented example demonstrates how the machine acts compliantly and intelligently with the human intentions.

The presented example shows as well that the overall performance and stability of the human–machine interaction as proposed in this work is good. This is assured by choosing the motor-schema approach together with a negative feedback control law. However, there are still no tools to analyze the stability and dynamic performance of a dynamic system modeled as dynamic Bayesian network. This is a challenging and important problem that should be investigated. Although the presented architecture is a general-purpose one, still some work should be done regarding its applicability and performance when dealing with even more complex domains.

Acknowledgements

This article is written while the author was with the Department of Artificial Intelligence at the University of Freiburg in Germany. His stay was sponsored by a research fellowship from the Alexander von Humboldt Foundation.

References

1. Bailey-Kellog, C. and Zhao, F.: Qualitative spatial reasoning: extracting and reasoning with spatial aggregates, *AI Mag.* (2003).
2. Bratman, M.: *Intentions, Plans, and Practical Reason*, Harvard University Press, Massachusetts, (1987).
3. Bratman, M.: *Faces of Intention: Selected Essays on Intention and Agency*, Cambridge University Press, United Kingdom, (1999).
4. Bui, H., Venkatesh, S. and West, G.: Policy recognition in the abstract hidden Markov model, *J. Artif. Intell. Res.* **17** (2002), pp. 451–499.
5. Burke, J., Murphy, B., Rogers, E., Lumelsky, V. and Scholtz, J.: Final report for the DARPA? NSF interdisciplinary study on human–robot interaction, *IEEE Trans. Sys., Man and Cybern, Part C* **34**(2), 2004, pp. 103–112.
6. Charniak, E. and Goldman, R.: A Bayesian model of plan recognition, *Artif. Intell.* **64** (1993), pp. 53–79.
7. Dennett, D.: *The Intentional Stance*, MIT, Massachusetts, 1989.
8. Heinze, C.: Modeling Intention Recognition for Intelligent Agent Systems, Doctoral Thesis, the University of Melbourne, Australia, 2003.
9. Inagaki, T. and Stahre, J.: Human supervision and control in engineering and music: similarities, dissimilarities, and their implications, *Proc. IEEE*, **92**(4), (2004) pp. 589–600.
10. Johannsen, G.: Auditory displays in human–machine interfaces, *Proc. IEEE*, **92**(4) (2004) pp. 742–758.
11. Kautz, H. A. and Allan, J: Generalized plan recognition, in: *Fifth National Conference on Artificial Intelligence*, Philadelphia, PA, (1986), pp. 32–38.
12. Lesh, N. and Edzioni, O.: Insights from Machine learning for plan recognition", in: *The Next Generation of Plan Recognition Systems: Challenges for and Insight from Related Areas of AI*, Montreal, Canada, (1995), pp. 78–83.
13. Lesh, N., Rich, C. and Sidner, C.: Using plan recognition in human–computer collaboration, in: *The Seventh International Conference on User Modeling*, Banff Canada, 1999, pp. 23–32.
14. Moratz, R., Fischer, K. and Thora, T.: Cognitive modeling of spatial reference for human–robot interaction, *Int. J. Artif. Intell. Tools*, **10**(4) (2001) pp. 589–611.
15. Mulder, F. and Voorbraak, F: A formal description of tactical plan recognition, *Information Fusion*, **4** (2003), pp. 47–61.
16. Murphy, K.: Dynamic Bayesian Networks: Representation, Inference, and Learning, Doctoral Thesis, University of California, Berkeley, 2002.
17. Murphy, K.: Bayes Net Toolbox for MATLAB, http://www.ai.mit.edu/~murphyk/Software/BNT/bnt.html, 2003.
18. Peng, H. and Ding, C.: Structure search and stability enhancement of Bayesian networks, *IEEE International Conference on Data Mining*, (2003).
19. Pynadath, D.: Probabilistic Grammars for Plan Recognition, Doctoral Thesis, the University of Michigan, MI, 1999.
20. Sheridan, T.: *Telerobotics, Automation, and Human Supervisory Control*, MIT Press, 1992.
21. Suzic, R.: Representation and recognition of uncertain enemy policies using statistical models, in: *NATO RTO Symposium on Military Data and Information Fusion*, Prague, Czech Republic, 2003, pp. 1–19.
22. Tahboub, K. A.: Natural and manmade shared-control systems: an overview, in: *IEEE International Conference on Robotics and Automation*, Seoul, Korea, (2001), pp. 2655–2660.
23. Wilensky, R.: *Planning and Understanding*, Addison-Wesley, Reading, MA, 1983.
24. Yedidia, J., Freeman, W. and Weiss, Y.: Understanding belief propagation and its generalizations, *International Joint Conference on Artificial Intelligence*, (2001).