

# Perspective Taking (Theory of Mind) and adaptive decision-making in HRI

Gazelle Zaheer, Maximilian Thorand and Qichao Xu

**Abstract**— Human-Robot Collaboration (HRC) is one of the most crucial parts of industry 4.0, since it is a possibility to combine the respective strengths of humans and robots in a joint task. This group project focuses on a scenario, in which a task is assigned to either the robot or human to pick up an object. While the robot is assumed to always function and never fail to grasp, the human might fail to do so and thus the robot may come in to support the human with certain behaviour. The Human model is designed with Markov Decision Process (MDP) which will create different actions in different states of the current human state. While the human might fail to succeed in grasping the object, robot will observe this, acknowledge, and act accordingly. This is done via Partially Observable Markov Decision Process (POMDP), since the robots needs to estimate the human state belief based on the environment it can observe.

## I. INTRODUCTION

Human-robot collaboration is a possibility to combine the respective strengths of humans and robots in a joint task. Humans with their cognitive capabilities are able to react to their immediate environment, such as defective components or changing parameters of parts and processes. The advantages of robots on the other hand are the capability to function in monotonous, dull work while maintaining accuracy and endurance. Furthermore, future robots will also learn to react with the environment, which is investigated in this project work.

This project tries to solve the problem the factories facing in warehousing and the work assignment between Robot and Human. The project starts with receiving a command of “an object with an ID is ready to be picked by Human/Robot”. The ID information gives the specifications of the object including: the size, the storage conditions, time for delivery, delivery means (delivery on site or delivery to an external address) etc. Initially when object arrives for picking and packaging there will be decisions according to the object parameters (size, weight, temperature) in order to identify who should pick the object (Human or Robot). If object should be picked by Human and he is not paying attention or failed to pick it up then Robot should take over the assigned object. So, the technical problem is to find a suitable algorithm that assigns the most suitable decisions for the robot regarding to observing Human beside the task of picking up the object and packaging it so that the entire process consumes less time and faults.

Developing an interface to generate decisions according to the object parameters is one of the objectives this project will solve. Furthermore human, robot and conveyor belt objects will be simulated using MORSE. Designing the robot decision making and collaboration with Human using POMDP will be

key to achieve a structure of hand in hand working of human and robot.

## II. RELATED WORKS

In the work of [1], Görür et al use a stochastic model to predict humans intentions and desires from the observed environment. A POMDP based human-robot shared planner is used to model the interaction between them. In our work, we basically follow the work of [1] and use two stochastic models, one for the modelling of humans behaviours, one for the robots to estimate human intentions. Also, we try to simulate the interaction in more difficult scenarios and build a general framework.

## III. METHODOLOGY

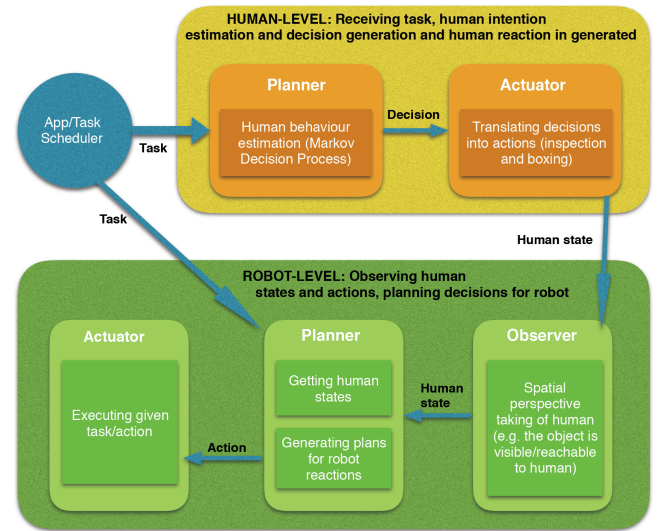


Figure 1: General Architecture

### A. General Framework

Our project main focus was human and robot collaboration not a simple interaction. Robot was not allowed to have direct interferes with human, both were doing their assigned tasks independently, just in case of help to human, there is an observer to report environment parameters including human and object pose and human actions. Therefore, there are two independent levels in project’s architecture where a separate node, “Task Scheduler”, connecting these two levels together.

Two levels are for Human and Robot. Human level consists of two nodes, Planner and Actuator. Robot level same as human level, but with an extra node, for observing human in case of help, which is Observer node.

Figure 1 shows general overview of system that is going to

solve the problem the factories are facing in warehousing. In following architecture three main parts considered which will be divided into smaller nodes. First part is “Task Scheduler” node which gets object's parameters and schedules tasks according to those parameters and finally sends a task whether to human or robot level. Second part is “Human Level” where human action of a given task will be estimated according to human behaviour in sub-node “Planner” using MDP modelling algorithms and a decision will be generated to next sub-node “Actuator” which will translate given decision into action and will execute it using MORSE simulator. Third part is “Robot Level” which consists of three sub-nodes, “Observer” which observes human actions and states using a mapper, then sends those observations to next sub-node called “Planner”. “Planner” will use those states to infer and generate plans for robot to act upon those plans. “Planner” node will use POMDP modelling algorithm since human states are uncertain each time. “Actuator” will receive actions from “Planner” node and will execute them, this node will use MORSE simulator to do mentioned tasks. Detailed explanation of architecture can be found in Appendix A of this report. In following next two sub-sections focus will be on Human and Robot level parts of the architecture.

### B. Human Level

To achieve establishment of efficient and smooth collaboration between Human and Robot, human should be considered as a special case study. Human beings are not predictable in their decisions and behaviours and in a factory where too much workload is on human that can make him tired or if he is a beginner in his work then he might not succeed in fulfilling tasks on time because he needs time to understand the workflow. There might be different use cases where robot will interfere in human tasks where robot should not do that and human needs to stop robot very safely and efficiently. Therefore, considering many use cases regarding to human collaboration with robot and in general human task execution in a factory, putting simulated human in our scenery and only manually controlling human from keyboard would not cover all use cases for both, human and robot. Modelling human can help us in covering all possible behaviours and decisions human might have in a factory and all possible use cases regarding to human and robot collaboration.

In general, human beings takes his future decisions and actions according to his current situation not his past situations. He might use his previous decisions as an experience but he can drop all his past decisions if he gets better decisions according to his current behaviours. Reinforcement Learning can help us designing our intended model, according to Alex J. Champandard in his article, Reinforcement Learning “allows the machine or software agent to learn its behaviour based on feedback from the environment. This behaviour can be learnt once and for all, or keep on adapting as time goes by.” [2] Describing such an environment for our human, Markov Decision Processes can be a good option. Markov Decision Processes or shortly MDPs as stated in this[3] article, it “provides a mathematical framework for modelling decision making in situations where

outcomes are partly random and partly under the control of a decision maker”. With MDPs we can design a fully observable environment where current state completely characterises the process.

MDP is a five tuple which consists of finite set of states, actions available from states, probabilities for taking available actions from states, immediate reward for transition from one state to another by action and discount factor for identifying importance between future rewards and present rewards. As mentioned in a teaching slides from David Silver[4] “The state captures all relevant information from the history, once the state is known, the history may be thrown away”. His paper helped in understanding MDPs model and design semi and fully autonomous model for our human. We designed a semi-autonomous model where we can have different tests and scenarios manually and can be useful for the real user tests. And also we designed a fully autonomous model for testing robot with rigorous tests and include randomness and an automated process to test like 1000 times, to achieve smooth collaboration.

Possible states for human who is working in a factory, depending on different types of human level experience (beginner, expert or distracted), can be:



Figure 2: Human states

Initially human will be assigned to a task, ( S0 ) which in our case will be grasping coming object from conveyor belt, checking it and putting inside box for further processes. Here we provide some use cases as an example for our semi-autonomous model. If human is an expert in executing given tasks then his possible states after assigning task state (S0) can be more Success (S1) and less Failure (S2) when he is Tired (S5) and wants to have some Recovery (S7). Another use case can be for beginner human, who is new to factory and tries to gain experiences and avoid mistakes because he can be fired. So for beginner human to do the task correctly, after task is assigned, most of the time he will first Evaluate (S4) given task description, or look around (S3) to see how others work, then attempt to grasp the coming object which might lead to Success (S1) or Failure (S2) due to not reachability of object or speed of conveyor belt might be fast for him to grasp it while he was evaluating or looking around. Distracted human in the other hand will most probably look around and not pay attention to his given task or most of the time leave the working space. Robot while observing beginner or distracted human might come to this belief that human might need help

## AAC-HRC: FINAL REPORT

in finding, reaching or grasping the object (in case of beginner human) and robot will try to point to object or grasp it which might lead to get warning from human because human was evaluating and then he wanted to do his task by grasping. In distracted human case, robot after grasp attempt can reach to success because object collisions could be occur and human will fail since he could not grasp the object. For semi-autonomous modelling where we will send manually a state for our human (to evaluate our different mentioned human types), possible use cases can have more than 500 scenarios while doing it without model would limit us to only few scenarios.

As we mentioned above for testing robot with completely random human behaviours and automatically, with no manual insertion of human states, we will design a fully autonomous model which will cover all human's possible states together and possible actions.

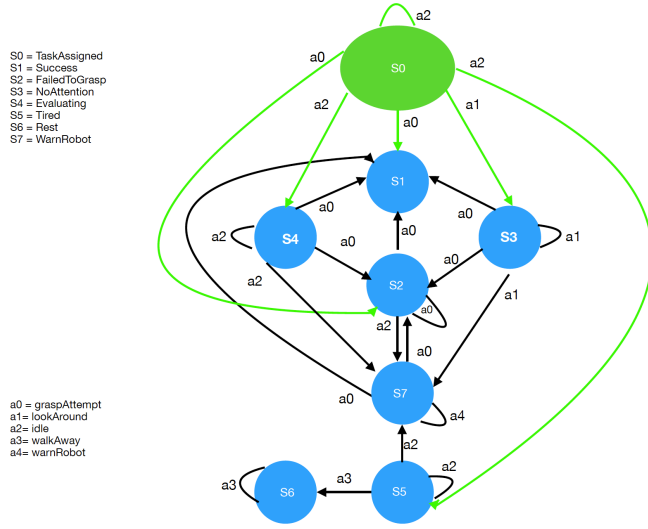


Figure 3: Human Model in MDP

### C. Robot Level

As stated in [5] PODMP will be the approach to estimate human state belief, since POMDP is able to deliver good results when it comes to estimating current state beliefs based on observation. Human model will produce actions based on its current state transitions and thus produce certain observations, which is the feed for the POMDP model and will hereby estimate the current human state belief and act accordingly. We need to do this, since the human is only partially observable, which means that robot can only estimate on the fact, in which the human currently might be in, because POMDP takes observation from its environment. Robot model is able to act in a stochastic way only, which means that he is driven by its environment. When the robot gets observations as an "input" he then can take action accordingly.

Firstly, the MDP model is overlaid, which means that human states are mapped to robot states, since robot can never be aware of real human states. Therefore, following POMDP model was created:

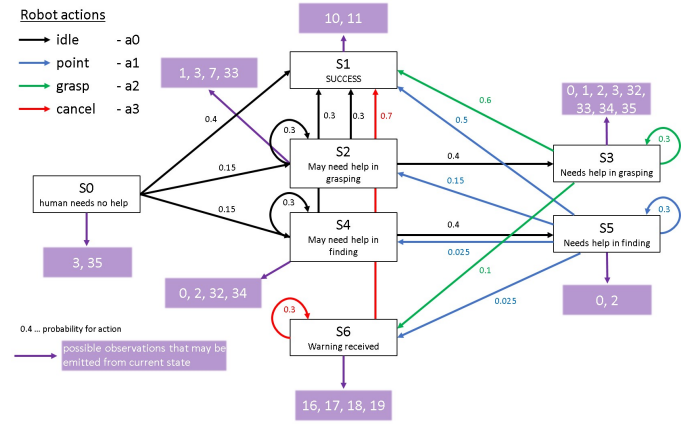


Figure 4: Robot Model in MDP with observations

The model describes the states for robot which are equal to the state belief of human. S0 indicates the initial state belief, which is reached, when the task is assigned to human. Robot will always take the action "idle" here, since the task was just assigned to the human and he assumes, that the human does not need help. While doing so, robot keeps observing the human and eventually make certain observations which will lead him into either state of "human may need help in grasping" or "may need help in finding", which set the robot into some kind of "awareness" state, where he does not take action yet. A short distraction will not lead into immediate action of robot, but the robot will be aware now. Further distraction of human or anything else not leading into the success of grasping the object will make the robot take action, but idle (idling is considered as a valid action). If the robot is entering state three or five, which indicate that a) human needs help in grasping or b) needs help in finding the object, robot will take action accordingly. This behaviour is enforced due to his rewarding system which will be explained to a later point. Finding the correct robot action based on its observations is key to lead the whole system to a success (state). Furthermore there is a warning "received state", which is implemented to prevent a harmful situation for the human. In some cases it might happen, that the robot wants to grasp the object, because he estimated the human not being aware of the situation or whatever reason. Despite a seemingly correct behaviour of the estimated state from robot, the human wants to grasp the object and will warn the robot to indicate, that it shall stop the action of grasping. This case is necessary to prevent any damage made to human, which implies possible physical contact of robot and human.

Observations are a crucial input for POMDP. Observations for this projects were done manually, which means that there is no external tool, to observe the environment. All observations are created in MORSE, which means that there are single environment parameters that are tracked (for instance "object is in range to human") and then summarised to a single observation. In this scenario, an observation can consist of 6 environment parameters which are

- Object is in range
- Object is visible
- (human) has object
- Grasp attempt
- Idle

#### - Warning received

The first 3 environment parameters indicate the status of the object itself and its current position in our world. Also, all actions from human can be observed, which are 1. Grasp attempt, 2.idling, 3.warn Robot, 4. Look around, 5. Evaluate. The human action “look around” and “evaluate” are covered by the environment parameters “object is visible” (ov), which means that when “ov” is not true, then look around or evaluate is always true and vice versa. With this, we have covered all possible observations our small world can produce by adding each of these environment parameters in its binary state and their combination together (for further details refer to the excel sheet in user manual or appendix B).

This approach is a very detailed way on how to get all possible observations of the scenario at hand, since there is no combination of environment parameters left out.

## IV. RESULTS

We considered testing human and robot first with three semi-autonomous models: Beginner Human, Distracted Human and Tired Human. After running each model for about ten times most of the time Beginner Human after many evaluation states and avoidance of robot interfere he reached the success but sometimes human instead of taking action of warning or grasping in WarningRobot state, he still was evaluating which leads to Failure since robot was grasping the object.

Distracted Human model, most of the time was looking around and even there were warning states where he had to follow grasping or warning but he was still looking around and robot most of the time pointed or grasped the object. There were few cases where human was taking action of grasp attempt in warning state after warning the robot to not interfere but only few of his grasp attempts were successful.

Tired Human, success state was not to grasp the object and reach actual Success state but instead when he was in Tired state so he was idle, doing nothing, and then walking away and staying idle in Recovery state. That's why this model was always successful because there were not that much complex state transitions. And robot always was executing grasping action after human is gone and is not returning back.

For semi-autonomous modelling where you are trying to adjust states and actions to lead agent's beliefs to your use cases, the only challenge is, the more your use cases have more complex transitions the more adjustment will lead to inaccuracy. Therefore, it is better to divide your use cases in smaller scenarios to have deeper testings of your robot.

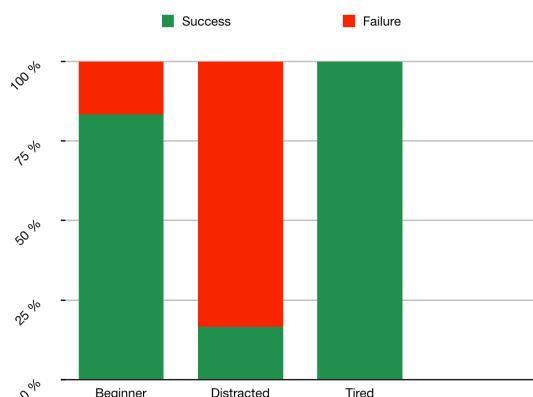


Figure 5: Human and Robot Interaction with three use cases

For fully-autonomous model problem of adjustment with agent's belief was not an issue since the model 100% adjusted to agent's belief. In this model adjusting actions rewards and probabilities were challenging, since randomness in actions and states and also these random actions and states to be relative to each other, were issues which had to be considered while designing and specially testing the model.

Following graph shows percentage of actions were taken randomly by agent(in our case human), since we considered beginner type human (expert would always succeed with only one action, grasp) where we putted more reward to evaluate before grasping. In graph, graspAttempt action has more percentage than evaluation since after evaluating human were taking action of grasping to reach success or sometimes after warning the robot he was attempting in taking action of grasping. Although our model would always start with evaluation but sometimes he was taking action of look around then grasp attempt. There were very less percentage and reward to look around action, giving beginner human this opportunity to look how other people are doing their task and then try to grasp. Therefore, according to our use case, fully-autonomous model was taking all actions as we designed, even it includes cases where human is tired and walks away.

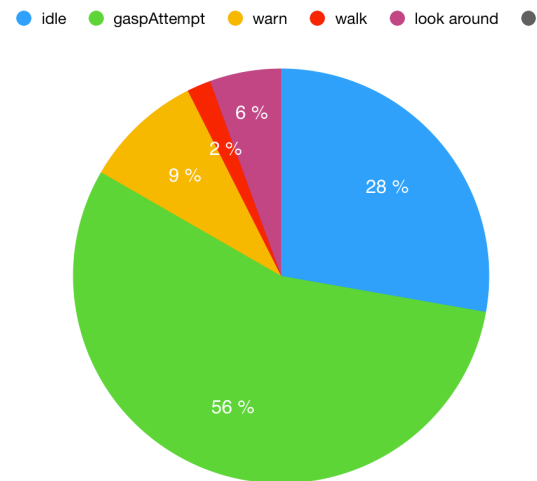


Figure 6: Randomness in actions

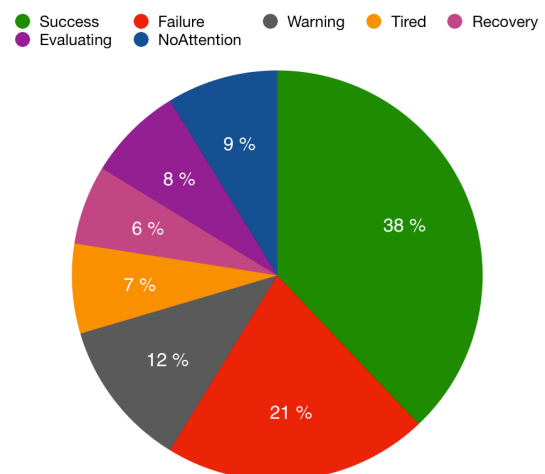


Figure 7: Randomness in states



Randomness in states according to our use case (beginner human), was almost appropriate. Although in graph you can see that success and failure have more percentage than other states, it is because each state (except Tired and Recovery) had transition to Success or Failure state. And reason that difference between failure and success is 17%, it is because we put 60% to grasp where it will lead to success and 40% to just do grasp attempt but not grasping to lead to failure state for robot reaction in human's grasp attempt action.

In general fully-autonomous model had almost relative random actions to random states. Meaning, human's any random action in each round, was according to state where we adjusted to be that way. For robot to follow those random actions and states from human was almost parallel to his observations, meaning if human was taking so long evaluation or looking around then robot was pointing to or grasping the object, or when human was warning the robot to not grasp then he was canceling his grasping action.

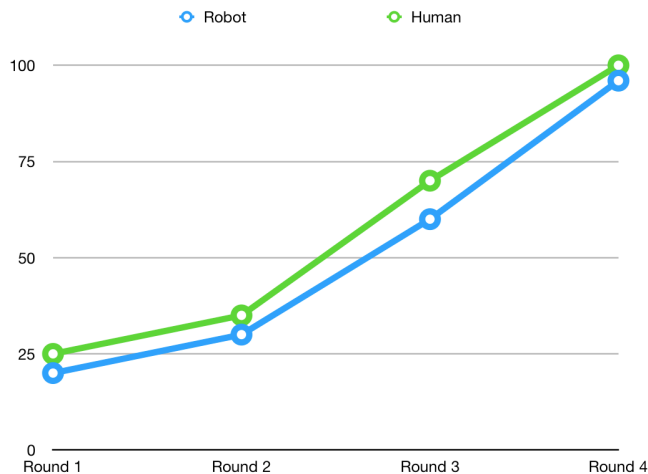


Figure 8: Human and Robot collaboration

Using fully-observable modelling (MDPs) can be very useful in testing robots who has interaction and collaboration with human, and it can be safer where we not only simulate robot for testing but we can also put simulated human to have different test cases with robot. With fully autonomous model we can test robot with rigorous tests and include randomness and an automated process to test like 1000 times. And with semi-autonomous model we can have different tests and scenarios manually which are necessary for the real user tests.

For POMDP modelling it is necessary to mention that the given approach taken is fairly optimised concerning getting observations. The model at hand therefore produces less errors and mistakes due to that.

## V.

## CONCLUSION

The collaboration of human and robot performing the task of grasping an object on a conveyor belt can be solved using MDP modelling for human and POMDP for robot. While doing so, observations are a crucial input for POMDP to work and estimate the current state belief of human.

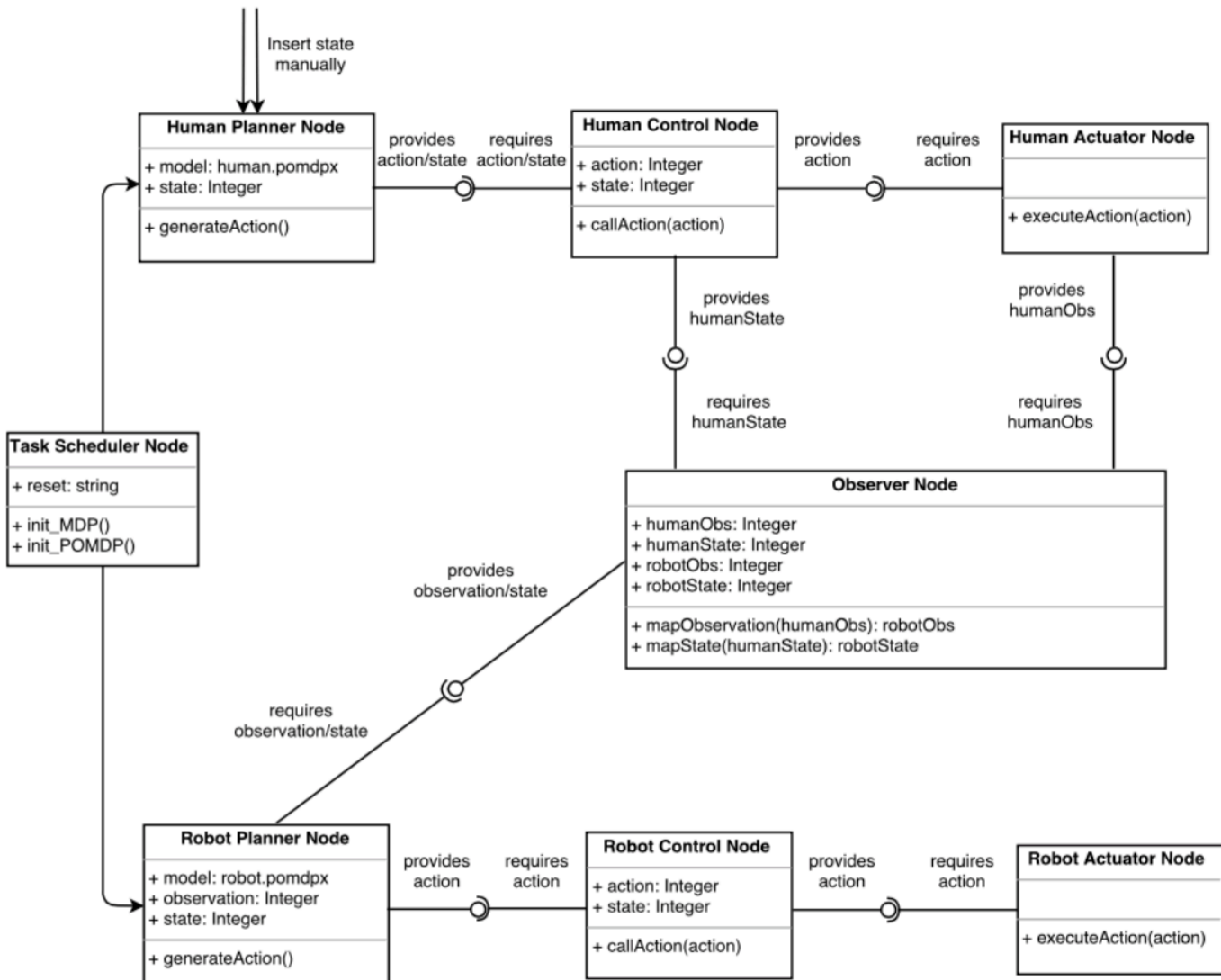
The project lead to positive results (21 out of 25 successfully tested scenarios – achieving success state) due to the combination of well examined observations and rewarding system for both, human and robot. A possible reason behind this might be the fact, that the observations are calculated manually by MORSE with algorithms, which always outputs correct observations, but in reality we can not assure this. In reality, observations can be misled by noise and other distractions to sensors and cameras of the robot.

Further investigation of this project work would be the implementation of a tool, which will output observations by itself, and not via code in MORSE. Also, different human types can be created and robot behaviour can be adjusted, since the estimation of state beliefs can always be improved.

## REFERENCES

1. G.rür O C, Rosman B, Hoffman G, et al. *Toward Integrating Theory of Mind into Adaptive Decision-Making of Social Robots to Understand Human Intention*[J].
2. Alex J. Champanand, "Reinforcement Learning," *Reinforcement Learning*, 2002. [Online]. Available: <http://reinforcementlearning.ai-depot.com/>.
3. J. L. Taolue Chen, "Towards Analysis of Semi-Markov Decision Processes," FMT, University of Twente, The Netherlands.
4. David Silver, "Lecture 2: Markov Decision Processes."
5. A.R. Cassandra. Exact and Approximate Algorithms for Partially Observable Markov Decision Processes. PhD thesis, Brown University, 1998

## Appendix A: Detailed System Architecture



## Appendix B: Short example of Binary Overview of Observations

Binary overview all possible observations										
# of observations	object visible	object in range	grasping attempt a0	has object	warning received a4	idle a2				omit reason
0	0	0	0	0	0	0	0			
1	1	0	0	0	0	0	0			
2	0	1	0	0	0	0	0			
3	1	1	0	0	0	0	0			
4	0	0	1	0	0	0	0			Logic
5	1	0	1	0	0	0	0			Logic
6	0	1	1	0	0	0	0			Logic
7	1	1	1	1	0	0	0			
8	0	0	0	0	1	0	0			
9	1	0	0	0	1	0	0			Logic
10	0	1	0	0	1	0	0			
11	1	1	0	0	1	0	0			
12	0	0	1	0	1	0	0			Logic
13	1	0	1	0	1	0	0			Logic
14	0	1	1	0	1	0	0			Logic
15	1	1	1	0	1	0	0			Logic
16	0	0	0	0	0	1	0			
17	1	0	0	0	0	1	0			
18	0	1	0	0	0	1	0			
19	1	1	0	0	0	1	0			