

# Human Intention Recognition for Human Aware Planning in Integrated Warehouse Systems

Tomislav Petković<sup>1</sup>, Jakub Hvězda<sup>2</sup>, Tomáš Rybecký<sup>2</sup>, Ivan Marković<sup>1</sup>,  
Miroslav Kulich<sup>2</sup>, Libor Přeučil<sup>2</sup>, Ivan Petrović<sup>1</sup>

**Abstract**—With the substantial growth of logistics businesses the need for larger and more automated warehouses increases, thus giving rise fully robotized warehouse shop-floors with mobile robots in charge of transporting and distributing goods. However, even in fully automatized warehouse systems the need for human intervention frequently arises, whether because of maintenance or because of fulfilling specific orders, thus bringing mobile robots and humans ever closer in an integrated warehouse environment. In order to ensure smooth and efficient operation of such a warehouse, paths of both robots and humans need to be carefully planned; however, due to the possibility of humans deviating from the assigned path, this becomes an even more challenging task. Given that, the supervising system should be able to recognize human intentions and its alternative paths in real-time. In this paper, we propose a framework for human deviation detection and intention recognition which outputs the most probable paths of the humans workers and the planner that acts accordingly by replanning for robots to move out of the human’s path. Experimental results demonstrate that the proposed framework with the human intention recognition module increases total number of deliveries by 9.2%, shortens order completion times by 0.8% and lowers the number of replanning by 38.7%.

## I. INTRODUCTION

With the considerable global expansion of e-commerce in the past years a strong demand for efficient automation in the warehouse industry is following suit. Given that, rapid processing of incoming orders and nonstop warehouse operation have become a paramount topic for companies having automated warehouses at the core of their businesses, e.g. Amazon and Swisslog, as well as for third-party logistics companies. Even though complete warehouse automation with a considerable number of mobile robotic platforms is at the heart of such systems, the need for human intervention and collaboration with the robots still exists; indeed, it can even be beneficial for productivity and competitiveness [1]. A simulated example of a typical automated flexible warehouse can be seen in Fig. 1. It consists of large storage racks, packed with goods that are carried by a fleet of mobile robots to designated picking stations, where humans pick goods,

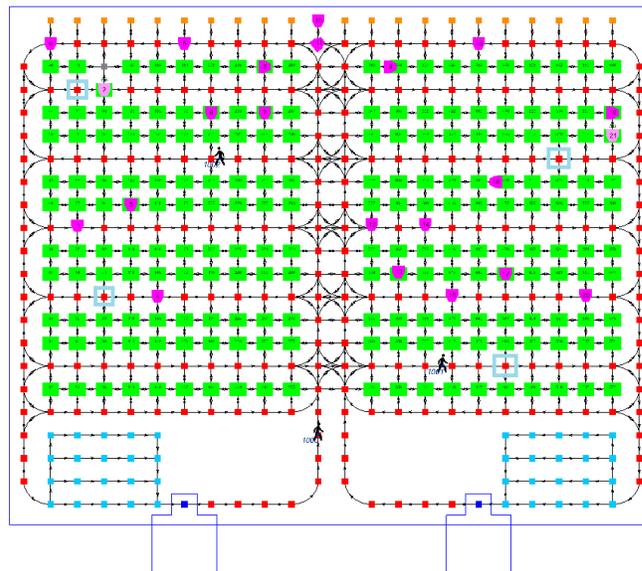


Fig. 1: The developed flexible warehouse simulator. Mobile robots (pink) do a predefined set of tasks such as carrying storage racks (green) while moving on the ground nodes (red). Three human workers move freely between the storage racks picking goods at specified warehouse locations or doing maintenance work.

pack them, and forward further for shipment. Furthermore, humans can enter the warehouse shopfloor and move freely among the robots. However, human workers do not always behave in a deterministic and prescribed fashion which can affect the carefully orchestrated coordination devised by the robot fleet management system (FMS); hence, to adapt to such perturbations, a human intention recognition (HIR) system is needed.

The task of the FMS, which carries out all the robot planning tasks, is to find trajectories between a pair of nodes in the warehouse while taking into account the plans of other robots. The problem of trajectory planning and motion coordination is a well studied problem [2], and since then many approaches have been introduced often based on the classical single-robot planning approaches [3]–[6] that provide completeness or even optimality; however, they are not practical for warehouse environments due to their large computational complexity. To counter this issue, another category of sub-optimal planning algorithms has been introduced [7]–[11], among which the context aware route planning (CARP) algorithm, in particular, provides good quality solutions in a warehouse environment with

<sup>1</sup> Tomislav Petković, Ivan Marković and Ivan Petrović are with the University of Zagreb Faculty of Electrical Engineering and Computing, Department of Control and Computer Engineering, Laboratory for Autonomous Systems and Mobile Robotics, Unska 3, HR-10000, Zagreb, Croatia (email: petkovic@fer.hr, ivan.markovic@fer.hr, ivan.petrovic@fer.hr).

<sup>2</sup> Jakub Hvězda, Tomáš Rybecký, Miroslav Kulich and Libor Přeučil are with the Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague, 160 00 Prague, Czech Republic (email: hvezdjak@fel.cvut.cz, rybectom@fel.cvut.cz, miroslav.kulich@cvut.cz, preucil@ciirc.cvut.cz).

low path computation time and is easily extendable with warehouse specific constraints. The main drawback of the original algorithm was its reliance on the ordering of the agents to be planned. Several heuristic approaches have been introduced in [12] to improve the properties of the algorithm. However, in the warehouse environment the tasks are not known all at the same time but are given sequentially.

As stated earlier, to ensure warehouse operation efficiency, FMS needs to be complemented with HIR, thus effectively enabling a *human aware planner* (HAP). Given that, when a deviation has happened, HAP should be notified and assisted with the estimation of paths the human might follow – this becomes possible if worker intentions are accurately recognized. In the current paper, our approach to HIR relies on the *Bayesian theory of mind* [13], where authors introduced a model for estimating student desires to eat at a particular food-truck by observing their motion. Examples of models using Markov decision processes can be found in [14], where authors proposed a framework for estimating pedestrian intention to cross the road, and in [15], where authors proposed a gesture recognition framework for robot assisted coffee serving. Learning based methods for trajectory estimation or HIR using motion pattern learning have also been studied in [16]; however, in [17] this approach was criticized where authors emphasize that such techniques operate offline and imply that at least one example of every possible motion pattern is contained in the learning data set, which often does not hold in practice. They propose using growing hidden Markov models for predicting human motion, a problem which we consider dual to the human intention estimation in the warehouse domain. In [18] a thorough review is of human intention recognition is given emphasizing also its potential applications in decision making theory, and in [19] authors give overview of human motion prediction methods.

In this paper we propose a robot fleet management system endowed with human worker intention recognition. The HIR module is based on the Bayesian theory of mind giving a prediction of the human trajectory, should it deviate from the assigned path, and the proposed system is capable of taking into account the human deviations without the necessity to cease the operations of all the robots during human presence inside the warehouse. The HAP reacts by moving the robots out of the human’s way or simply stopping the robots. Even though we assume that robots are equipped with a safety system, human aware planning can reduce the chance of robots driving close to humans thus lowering the number of times the safety system is triggered. In conjunction, this leads to more efficient operation of the warehouse and hypothetically less stress on the human workers (note that load carrying robots can weigh up to 800 kg). We have benchmarked the proposed HIR framework with other methods and recorded 5.8% increase in total deliveries and 4% reduction of average delivery time.

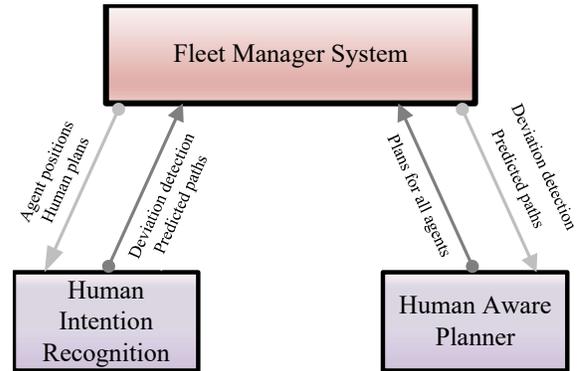


Fig. 2: The proposed system architecture. HIR and HAP are run as separate threads of the FMS. Note that we have highlighted only the data flow which we describe in the scope of this paper.

## II. PROPOSED HUMAN AWARE ROBOT FLEET MANAGEMENT SYSTEM

One of the main problems warehouse management currently faces is that most issues inside the warehouse shopfloor, e.g., a robot malfunction or goods falling on the floor storage racks, require human attention. Furthermore, sometimes it is more efficient for a human worker to carry out the picking of goods if their distribution in the warehouse is too disparate. Given that, there are many times a human intervention in the robotized warehouse is needed, and if this is not carried out in a planner manner it can lead to interruption of the whole warehouse operation, ultimately exacerbating the issue we were trying to solve in the first place.

The proposed system architecture is shown in Fig. 2. FMS is in charge of planning paths for all the robots and humans as well (agents in short). The HIR module serves as part of the FMS with the task of assisting the HAP. HIR has at its disposal information about the position of all the agents in the warehouse, as well as planned paths of human workers. While the robots are controlled by FMS and always follow the paths given to them, human workers can deviate from their paths for a number of reasons. The job of the HIR module is to detect human deviations from the planned path and estimated the predicted trajectories, based on which the HAP will produce an updated plan for all the agents. To efficiently assist the HAP, HIR should ascertain (i) if a worker deviated from the original path and (ii) where that particular worker is going to. Given that, output of the HIR module is a logical flag indicating if there is a worker deviation, followed by a set of probable paths. The HAP then reacts with the method described in Section II-C. In the sequel, we describe each of the proposed system’s components in details.

### A. The warehouse simulator

The simulated warehouse shown in Fig. 1 is organized into connected nodes and resembles faithfully the software architecture used by an FMS of a true robotized warehouse systems. Each node can either be occupied by a robot, storage

rack or a human. Unloaded robots can move across all free nodes, while loaded robots cannot enter a node already containing a storage rack. As incoming orders arrive, the robot fleet management system coordinates all the robots so that storage racks containing ordered goods are delivered to picking stations, then returned to a free storage node (not necessarily the same as the starting one), and the robots that are idle are sent to the charging nodes. All this needs to be carried out in an efficient manner ensuring continuous operation of the whole warehouse. This planning task is quite challenging since the number of robots can range from 50 to 800, and it gets an additionally layer of complexity by having to account for human workers in the area. Note that we assume that robots are equipped with a safety system ensuring that the robots will stop if they come to a range that is too close to the human worker. The multi-robot warehouse simulator was originally presented in [20], while for the current paper we have extended the simulator with the ability to include human worker plan deviation.

### B. Human intention recognition

1) *Human Deviation Detection*: Each worker that enters the warehouse has a predefined path that consists of a sequence of ground nodes shown in Fig. 1. We designate the first node of the worker path as the *current node*, and the second node of the path as the *next node*. Every time the distance between the worker and the next node in the path is less than  $r = 0.25$  m, that node becomes worker's *current node* and its successor in the path sequence becomes worker's *next node*, until the end of the path is reached. Human beings usually do not walk in a perfect straight line [21], but swing laterally while moving forward. Given that, we allow deviation from the path defined by the *allowed deviation area* that is described by an ellipse having focal points in the current node and next node, while the major axis is defined as the Euclidean norm between the focal points increased by  $2r$ . The allowed deviation area is shown in Fig. 3 and the worker is considered to be deviating from the path if it has been outside of the area for at least 4 consecutive cycles.

2) *Human Path Prediction*: If a worker is detected to be deviating from its predefined path, it is necessary to estimate its future path. In this section we propose a human path prediction method relying on our recent result [22], where we proposed a human intention estimation method based on hidden Markov model (HMM) motion validation. We describe it here briefly and direct the reader to the original paper for details.

We assume that there is a finite number of possible goal locations, which are usually in front of the storage racks of interest. For current experiments, we selected four auxiliary goals, each one in a different corner of the warehouse and their locations are labeled with turquoise rectangles shown in Fig. 1. It is important to emphasize that the worker is not required to go to the predefined goals, but they do serve as starting points for the proposed algorithm. Also, the last

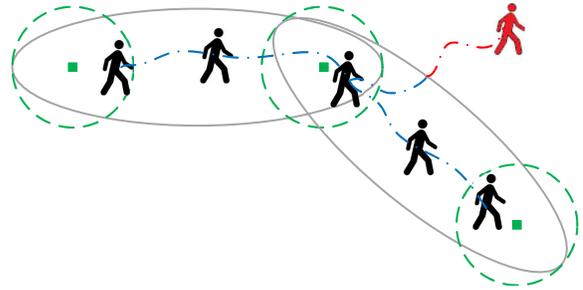


Fig. 3: The allowed deviation area is depicted by a grey ellipse defined by the current node and next node as focal point (green). Once the worker enters the green circle surrounding the next node, it becomes worker's current node and its successor in the path sequence becomes worker's next node. The example of human path which follows the plan is given with blue color while the path deviating from the plan is red.

node of the human's path provided by the planner is also considered a goal location.

Before the simulation starts, we calculate the distance between all the ground nodes using the D\* algorithm [23] and save it in a distance matrix  $\mathbf{F}$ . In case of a robot blocking the edge between two nodes during the experiment, we discard that edge from the graph and recalculate the distance matrix  $\mathbf{F}$ . Because we use road nodes, the search space is reduced significantly and the recalculation can be made in  $3.456$  ms<sup>1</sup> for 228 nodes and 348 edges of the warehouse road graph. For comparison, if we used a grid map representation with the precision of 10 cm, the recalculation would be done on approximately  $2 \times 10^7$  nodes and  $1.5 \times 10^8$  edges which would make the recalculation time larger than one minute, thus rendering it too long for real-time application.

Each time a worker makes a significant displacement, we update its predefined goals intention estimate using a scaled down version of the algorithm proposed in [22]. We associate the position of the worker with the observable nodes by forming a so-called *association vector*  $\mathbf{c}$ . The closer the human is to the node, the larger the value of the vector  $\mathbf{c}$ . By multiplying  $\mathbf{c}$  and  $\mathbf{F}$ , and by isolating the goal nodes, we obtain a modulated distance vector  $\mathbf{d}$  of dimension  $g$ , where  $g$  is the number of goals. We also calculate the alternative association vector  $\mathbf{c}'$  of the positions the worker might have gone to, if it moved the same distance from the last observation; we also calculate the corresponding modulated distance vector  $\mathbf{d}'$ . By comparing values of the vector  $\mathbf{d}$  with values of each  $\mathbf{d}'$  that we collect in matrix  $\mathbf{D}$ , we calculate the observation vector  $\mathbf{o}$  via element-wise division:

$$\mathbf{o} = \frac{\max_{1 \leq i \leq n} \mathbf{D}_{ij} - \mathbf{d}}{\max_{1 \leq i \leq n} \mathbf{D}_{ij} - \min_{1 \leq i \leq n} \mathbf{D}_{ij}}. \quad (1)$$

If the worker is moving towards a goal, the corresponding value of  $\mathbf{o}$  will be close to unity, and if it is moving away from that goal, the corresponding value will gravitate to zero.

<sup>1</sup>Configuration used for testing: Intel@Core™i7-7700HQ CPU @ 2.80GHz×8 with 15,5 GiB memory

We record the observation history and process it with an HMM with  $g + 1$  states, one for each goal and one for the last node of human’s predefined path. We define the HMM’s transition matrix  $\mathbf{T}$ :

$$\mathbf{T} = \begin{bmatrix} \alpha & \frac{1-\alpha}{g} & \dots & \frac{1-\alpha}{g} \\ \frac{1-\alpha}{g} & \alpha & & \vdots \\ \vdots & & \ddots & \\ \frac{1-\alpha}{g} & \dots & & \alpha \end{bmatrix}, \quad (2)$$

with the probability of staying in the current state  $\alpha = 0.823$ . We have obtained this parameter by learning on the recorded data with worker moving in the simulated warehouse without robots and minimizing the average displacement error [24]. Using this formulation of  $\mathbf{T}$  we allow the worker to change its mind of going towards any of the goals during the experiment. Finally, we set initial probabilities of worker’s intentions as:

$$\Pi = [g^{-1} \quad \dots \quad g^{-1}], \quad (3)$$

indicating that all of the goals are equally probable. During the experiment we use the Viterbi algorithm [25] to output probabilities of the worker going to each goal, which we consider as intention estimations.

After recording the probabilities of each goal we query if the probability of the worker going to the last node of the human’s path is high enough by comparing it to the largest of the probabilities. If their difference is less than the threshold of  $0.25g$ , we assume that the worker still might be going to the original goal and we report it to the HAP. Otherwise, we find all goals with the probability higher than the threshold of  $0.8g^{-1}$  and using the D\* algorithm on warehouse nodes shown in Fig. 1, we find the shortest path towards these goals on the road nodes. These paths are then reported to the HAP. For more detailed results on the path prediction we direct the reader to previous work [26].

### C. Fleet Management System

1) *Multi-robot route planning*: In this section we leverage the planning method we proposed in [20] that is based on the CARP algorithm [7]. The original algorithm structures its map as a resource graph, where each resource has a corresponding timeline. This timeline consists of free and occupied time windows, that indicate whether the resource is available in a given time interval or already taken by a different agent. For each agent that needs planning, the algorithm then finds free time window on the corresponding resource to the start node of the agent and uses a modified A\* algorithm to find the shortest path to free time window on the corresponding goal resource through expansion to the neighboring overlapping time windows. The advantage of the used approach is that it takes into account only handful of the most influencing agents. The algorithm aims to generate a trajectory for an agent  $a_k$  while assuming that trajectories for  $k - 1$  agents are already planned. This can lead to modification of those planned trajectories to accommodate the new agent. The main idea is that the algorithm iteratively

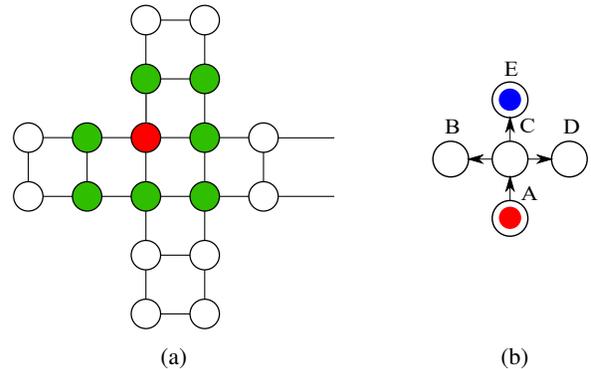


Fig. 4: (a) An example of conflicting states. As long as the robot is standing on the *Red* node, no other robot can stand on any of the *Green* nodes. (b) An example of planning with conflicting states. *Blue* robot wants to go to node *B*, and *Red* robot stays on node *E*. The node *C* is in conflict with nodes *B*, *D* and *E*. This means that when *Blue* robot attempts to plan to node *B*, the planning fails because even though the *physical* timeline of node *C* is free, the *conflicting* timeline of node *C* shows that it is occupied by the *Red* robot. The resulting merged timeline then shows it is not possible to go through node *C*.

builds a set of agents whose trajectories affect the optimal trajectory of agent  $a_k$  the most and replan this set of agents as well as agent  $a_k$  in ordering that yields solution with the best global cost.

Another consideration for planning in a warehouse environment is the presence of several types of constraints such as two states of two robots being mutually exclusive, e.g., robot  $a_m$  being on node *A* with rotation  $x$  is mutually exclusive with any other robot  $a_n$  being on node *B* with rotation  $y$ . Another type of constraint is the inclusion of the safety regions around the robot that differ in their radius and the interaction with robots. There are three different safety regions defined, but only two of them are considered for planning:

- Safety region 1: Robot stops its operation if it enters this radius
- Safety region 2: Planner avoids planning robot in this region
- Safety region 3: Robot must decrease its speed.

As the reader can notice, the safety regions 1 and 2 are identical for the planning algorithm, because the planner needs to avoid planning robots into them, and therefore the algorithm considers only regions 2 and 3.

To accommodate the constraints, the planning algorithm adds an additional timeline to each resource on the resource graph. This timeline is referred to as *conflicting*, with the first main timeline is referred to as *physical*. The *physical* timeline indicates when the resource is being physically occupied by an agent, while the *conflicting* timeline indicates whether any resource that the current resource conflicts with is being physically occupied. An example of conflicting states can be found in Fig. 4a, while an example of planning with conflicting states is shown in Fig. 4b. If these timelines were

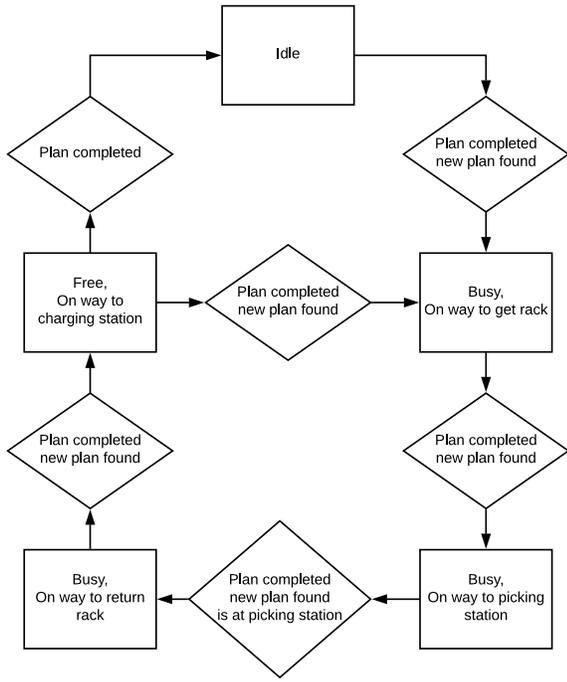


Fig. 5: The main state robot diagram

merged into one, the result would be the timeline where the free time windows are windows that the agent can move into without violating any constraints.

The safety regions for humans are handled similarly, i.e., by keeping timelines *Region 2* and *Region 3* for each resource and human present in the warehouse, corresponding to time windows that the human plan occupies. Furthermore, the *Region 2* timeline is added for each robot as well. The *Region 2* is used by merging the *physical* and *conflicting* timelines to obtain the final occupancy timeline of a resource during robot planning. *Region 3* occupancy timeline is used to check if the resource intersects the Safety region 3 during the computation of the time it takes the robot to cross a given resource.

2) *Robot path planning*: The proposed planner differentiates five different states where a robot can be *i*) going to pick up its rack assigned by the job, *ii*) taking the rack to the start of the queue before the goal picking station, *iii*) in the picking station queue, *iv*) heading back to return it to its position or *v*) heading back to its charging station. In addition to these planning states, the robot also has five internal states: *Idle*: The robot is at the charging station, *Busy*: The robot has a job assigned, currently working on its completion, *Free*: No job assigned, returning to the charging station, *Interrupted*: The state invoked by human action; the robot has been interrupted and needs to be replanned and *Failed*: The robot failed to find a plan; switched to *Interrupted* every few seconds. The detailed diagram of the standard robot operation without human interruption is shown in Fig. 5.

#### D. Human Aware Planner

1) *Planning for humans*: We assume that the worker is equipped with a system, such as a hand-held screen or augmented reality glasses, that can navigate the human

through the warehouse. The planning is done in a similar manner as the robot planning; however, the human always takes precedence to robots in the planning process. When the human planning starts, all the robots are interrupted. Once all the robots have stopped, the planner attempts to find a path to the human goal destination, while considering stopped robots as obstacles and taking into consideration the *Safety region 2* region where robots should not enter. If such a path is found, it is returned by the system for the human to follow, and the system automatically replans the interrupted robots, while taking the human plan into account. However, if such a path does not exist the system attempts to move the robots out of the way by first planning the human to the goal node, while not taking any of the robots into consideration. The robots then attempt to plan their paths to the closest possible node to their current goal, that is not in conflict with the human path. If the paths for all robots are found, it means that the evasive maneuver is possible and all the paths for human and robots are returned. If none of these approaches succeed, then the system indicates that the planning was unsuccessful. Moreover, to take into account the variance of human velocity, the system also plans the path for the human while taking into account the minimum and maximum velocity. Each resource along the human path is taken for a time interval  $w_i^r$  that starts at an entry time  $t_{entry}^{fast}$ , the time that would take the human to get to the goal if walking at maximum speed, and ends at the time  $t_{exit}^{slow}$ , the time it would take the human to leave if walking at minimum speed. Each time window  $i$  for all resources  $r$  in the path sequence is then  $w_i^r = \langle t_{entry}^{fast}, t_{exit}^{slow} \rangle$ .

2) *Planner reaction to the HIR input*: As described in Section II-B.2, once the human deviates from the planned path, the HIR determines paths to all the predefined goals whose probabilities exceed a given threshold. Once the planner registers paths from the HIR module, it interrupts all the robots and finds the longest common path segment of the obtained paths. This longest common path segment is then processed by the planner. Notice that if the planner took into account all the paths, it would possibly block a large portion of the warehouse. Once the segment that the planner will use for the human replanning is known, the system takes the first and last node as start  $s$  and goal  $t$  locations respectively. The planner then attempts to find a plan from the start location  $s$  to the goal location  $t$  using the nodes that were present in the planning segment  $\mathcal{T}$ . If found, the system updates the human path. If no such path is found, it means that some robot must be standing either directly on the segment of nodes or in a conflicting position. The planner attempts to find a plan to the goal location  $t$  in such situation but without the use of any specific nodes. If this planning succeeds, the system tells the human that he deviated from his original path and informs him of the new plan. However, if the planning is not successful, the system tells the human to stop immediately.

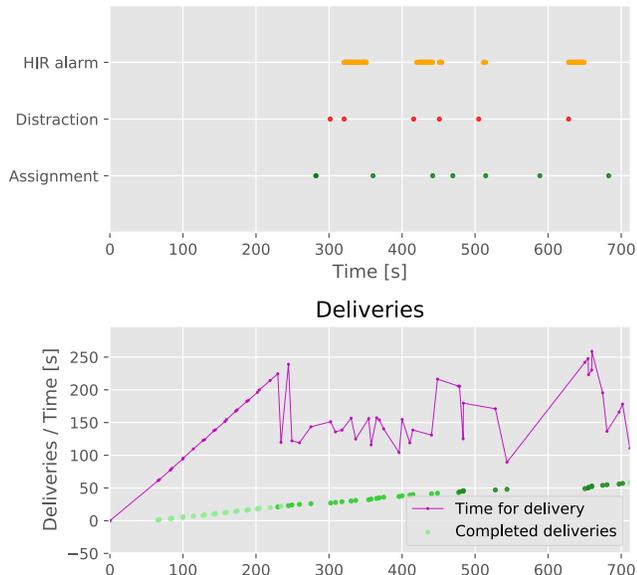


Fig. 6: Performance of the system with proposed HIR in one of the experimental setups. On top figure one can see times at which human is given assignment (green), when it deviated (red), as well moments in which proposed method signals deviation and sends path prediction to the planner.

### III. EXPERIMENTAL RESULTS

To demonstrate the functionality of the system, we have designed experimental setup with several delivering scenarios. All agents have set of assignments, *e.g.* picking or maintenance, that need to be completed during the experiment. We have benchmarked the average delivery time, total number of deliveries and average replan numbers if *i)* human does not deviate from the plan (ND), *ii)* human deviates and planner reacts using simple HIR module (SHIR), *iii)* human deviates and planner reacts using proposed HIR module (PHIR). The SHIR module outputs human path prediction with minimal change in heading assuming constant velocity. We have also tested the system with human deviations and planner reacts without HIR module (NHIR) and replanning is done only when human enters *Safety region 1* but we have omitted its analysis because it often does not allow human to complete its assignments until the end of experiments.

We have given the human assignments and simulated deviation multiple times during each experiment with random locations which are not known to the HIR module. The HIR module reacts promptly with alarm and path prediction which is then handled by the HAP, which can be seen in Fig. 6. Once the path is accepted by the HAP and human starts following it, the HIR alarm is turned off.

The results of 10 experiments with unique job sets for robots each lasting 750 seconds can be seen in the Table I. One can see that proposed method increases the total number of deliveries by 9.2%, order completion times are shortened by 0.8% and number of replanning is lowered by 38.7% when compared to the system with the simple intention recognition module which makes warehouse more efficient. The system without deviation obviously has the best results because replanning is needed only when the new human

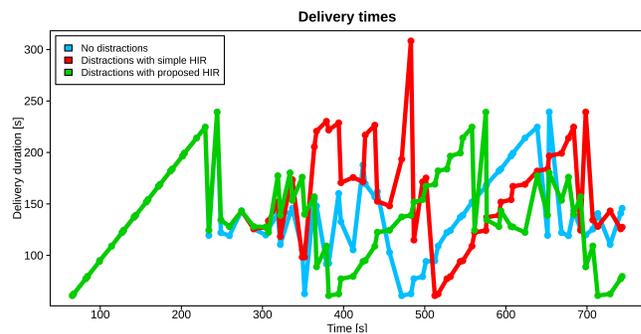


Fig. 7: Test results. The time needed for each delivery is labeled with filled circle. The human enters the warehouse at  $t = 250$  seconds and deviates at  $t = 300, 320, 420, 450, 500, 625$  seconds experiment time.

assignment arrives.

Example of one test run can be seen in Fig.7 where we compared delivery times for all methods. One can see that all methods perform the same while there is no human in the warehouse. Once the human enters, the delivery times are similar for each method. After the human deviates for the first time (300 s in the experiment), delivery times for next few deliveries are significantly lower for proposed method than for other methods. Because of that robots are able to receive new tasks before the end of the experiment which increases total number of deliveries. One can see that deliveries which system with HIR makes from 400 s to 560 s, are delivered much later for systems with simple HIR or no HIR (490 to 700 s). After 560 s the system with proposed method completes robot assignments that other methods were completing from 400 s to 560 s much faster. This causes increased number of total deliveries.

	ND	SHIR	PHIR
Average Delivery Time [s]	150.1	162.9	161.6
Total Deliveries	63	53.2	58.1
Average Replan Number	8.8	33.3	20.4

TABLE I: Experimental results.

### IV. CONCLUSION

In this paper we have proposed a framework of human intention recognition for human aware planning in integrated warehouse systems. The framework detects humans deviating from assigned paths and based on an HMM approach identifies and outputs the most probable paths that human worker is about to take. This information is fed to the human aware planner that is able to account for such deviations and replan the paths of warehouse robots so that efficiency is kept and encounters with humans are lowered (which slow down the process, since robots need to stop for a while due to safety reasons). We have conducted experimental runs on an in-house developed simulator and demonstrated that with human intention recognition the total number of deliveries increases by 9.2%, order completion times are shortened by 0.8% and number of replanning is lowered by 38.7% when compared to the system with the simple intention recognition module.

## REFERENCES

- [1] International Federation of Robotics (IFR), "The Impact of Robots on Productivity, Employment and Jobs," 2017.
- [2] L. E. Parker, "Path Planning and Motion Coordination in Multiple Mobile Robot Teams," *Encyclopedia of Complexity and System Science*, 2009.
- [3] M. R. Ryan, "Exploiting Subgraph Structure in Multi-Robot Path Planning," *Journal of Artificial Intelligence Research*, pp. 497–542, 2008.
- [4] J.-C. Latombe, "Robot Motion Planning," Jul. 1991.
- [5] S. M. Lavalle, "Rapidly-Exploring Random Trees: A New Tool for Path Planning," *Technical Report (Computer Science Department, Iowa State University)*, vol. 11, 1998.
- [6] M. Phillips and M. Likhachev, "Sipp: Safe interval path planning for dynamic environments," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 5628–5635.
- [7] A. W. ter Mors, C. Witteveen, J. Zutt, and F. A. Kuipers, "Context-aware route planning," in *Multiagent System Technologies*, J. Dix and C. Witteveen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 138–149.
- [8] D. H. Kiril Solovey, Oren Salzman, "Finding a needle in an exponential haystack: Discrete rrt for exploration of implicit roadmaps in multi-robot motion planning," *Algorithmic Foundations of Robotics XI*, pp. 591–607, 2014.
- [9] M. Peasgood, C. M. Clark, and J. McPhee, "A Complete and Scalable Strategy for Coordinating Multiple Robots Within Roadmaps," *Robotics, IEEE Transactions on*, vol. 24, no. 2, pp. 283–292, Apr. 2008.
- [10] M. Cap, P. Novak, A. Kleiner, M. Selecky, and M. Pechoucek, "Prioritized Planning Algorithms for Trajectory Coordination of Multiple Mobile Robots," *IEEE Transactions on Automation Science and Engineering*, vol. Special Is, 2015.
- [11] J. Hvězda, M. Kulich, and L. Přeučil, "Improved discrete rrt for coordinated multi-robot planning," *arXiv:1901.07363*, 2019.
- [12] A. ter Mors, "Evaluating heuristics for prioritizing context-aware route planning agents," in *2011 International Conference on Networking, Sensing and Control*, April 2011, pp. 127–132.
- [13] C. L. Baker and J. B. Tenenbaum, "Modeling Human Plan Recognition Using Bayesian Theory of Mind," *Plan, Activity, and Intent Recognition: Theory and Practice*, pp. 177–204, 2014.
- [14] T. Bandyopadhyay, K. S. Won, E. Frazzoli, D. Hsu, W. S. Lee, and D. Rus, "Intention-aware Motion Planning," in *Springer Tracts in Advanced Robotics*, vol. 86, 2013, pp. 475–491.
- [15] H.-i. Lin and W.-k. Chen, "Human Intention Recognition using Markov Decision Processes," *International Automatic Control Conference (CACs)*, pp. 340–343, 2014.
- [16] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 961–971.
- [17] D. Vasquez, T. Fraichard, and C. Laugier, "Growing Hidden Markov Models: An Incremental Tool for Learning and Predicting Human and Vehicle Motion," *The International Journal of Robotics Research*, vol. 28, no. 11-12, pp. 1486–1506, 2009.
- [18] T. A. Han and L. M. Pereira, "State-of-the-art of Intention Recognition and Its Use in Decision Making," *AI Communications*, vol. 26, no. 2, pp. 237–246, 2013.
- [19] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," *arXiv preprint arXiv:1905.06113*, 2019.
- [20] J. Hvězda, T. Rybecký, M. Kulich, and L. Přeučil, "Context-aware route planning for automated warehouses," in *International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 2955–2960.
- [21] C. E. Bauby and A. D. Kuo, "Active control of lateral balance in human walking," *Journal of biomechanics*, vol. 33, no. 11, pp. 1433–1440, 2000.
- [22] T. Petković, D. Puljiz, I. Marković, and B. Hein, "Human intention estimation based on hidden markov model motion validation for safe flexible robotized warehouses," *Robotics and Computer-Integrated Manufacturing*, vol. 57, pp. 182–196, 2019.
- [23] A. Stentz, "Optimal and Efficient Path Planning for Partially Known Environments," *Intelligent Unmanned Ground Vehicles*, no. May, pp. 203–220, 1997.
- [24] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *IEEE International Conference on Computer Vision (ICCV)*, 2009, pp. 261–268.
- [25] G. D. Forney, "The Viterbi Algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.
- [26] T. Petković, J. Hvězda, T. Rybecký, I. Marković, M. Kulich, L. Přeučil, and I. Petrović, "Human motion prediction framework for safe flexible robotized warehouses," in *IEEE International Conference on Robotics and Automation (ICRA2019), Long-term Human Motion Prediction Workshop*, 2019.