Altering and Improving Kiva

Some suggestions for improvement of the current Kiva system

Marcel Flipse
1473379
2011

Vrije Universiteit, Artificial Intelligence Department,
De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands,
marcel.flipse@gmail.com
Course: Bachelor Referaat , supervisor: Zhisheng Huang

Abstract. Kiva has improved overall performance of distribution centers by implementing robotics and concepts from Artificial Intelligence. Although the results look promising, there is still some room for improvement. Small changes in the environment or robot behavior rules might improve performance once more. On the other hand results of fundamental changes like introducing chaos are much more uncertain. This paper sketches some changes that could improve the performance of the current Kiva systems.

1 Introduction

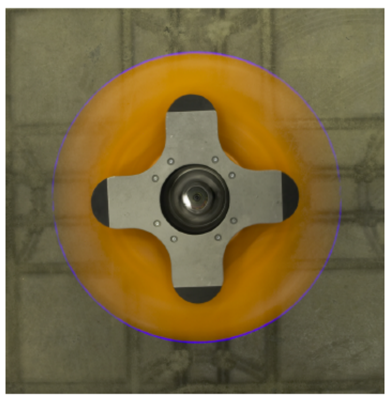
This paper starts with a short introduction on the Kiva system, a definition of some basic concepts and an explanation of control methods. After the introduction some changes to the Kiva system are given. Readers should have an interest in automated warehouses and robotics. Basic knowledge of computer systems might be useful.

1.1 Kiva

The Kiva system [9, 2] is a solution for automating pick, pack and ship orders used in distribution centers. It takes over product search and product retrieval assignments from human workers. Instead of applying technology to fit robots in an existing system, Kiva uses a custom environment.The floor of the distribution center is equipped with a barcode grid and small orange robots that fit underneath special product racks. A central server sends a message to a particular robot to perform a product retrieval task. This message contains the coordinates on the grid, which are barcode-stickers and information which specific sub part of the rack contains the needed product. The robot is equipped with a barcode laser so it can read its current position. The robot then calculates a path by combining its current location with the product location mentioned in

(a)

(b)



(c)

(d)

Fig. 1: a) A Kiva robot b) a Kiva warehouse implementation c) a Kiva drive unit rotating underneath the lifting device [2] d) a Kiva station [9]

the received message and moves through the environment by making 90 degree turns on specific coordinates. When the robot arrives at the product location, the robot then lifts the product rack, which can contain multiple different products, from the ground with a screw mechanism and transports the product rack to a station where a specific product is packed by a human worker to complete a customers order. When a robot arrives at a station the product space on the rack is highlighted with a laser so the human worker can quickly get the specific product. The human worker than scans the specific product and another laser highlights the box belonging to that specific order. After scanning the specific box the robot stores the product rack or uses the rack to fulfill another order at another location. Productivity is increased by saving search and retrieval time of products. It is more flexible than existing solutions like “Automated Storage and Retrieval systems”, where products are retrieved by conveyor belt systems or autonomous forklifts placed on rails. The Kiva shows scalability when the warehouse size increases only more robots need to be added instead of building a complete new system. Robustness is shown when a single robot breaks down. Only that specific robot stops working and not the complete system. Other non technical maintenance issues like cleaning the barcode readers on robots and replacing barcodes on the grid can be done quickly, while repairing and maintaining a conveyor belt system is a time consuming and expensive job.

1.2 Systems

Before addressing subjects like “Can the Kiva system be used for automated forklifts”, “Is Kiva a centralized or decentralized system” and “What can be improved in the current Kiva System”, some background information is needed.

Multi Agent Systems Kiva can be called a multi Agent system [8]. A multi agent system contains individuals that can be called agents when they comply with the following requirements:

1. an agent has control over its own actions; autonomy
2. an agent can interact with other agents; social ability
3. an agent can perceive its environment; reactivity
4. an agent exhibits goal directed behavior; pro-activeness
5. an agent is able to move around in its environment; mobility
6. an agent won’t communicate false information
7. an agent does not have conflicting goals; benevolence
8. an agent tries to achieve its goals; rationality

List items 1 to 4 are called the weak notion of agency. Many software agents would also comply with these properties, so this paper prefers to use the strong notion of agency which contains all list items. The only decision making in Kiva, that is not done by the robots, is the resource allocation. A central system selects which robot gets a specific retrieval order. Kiva robots are not only equipped with barcode readers but also with pressure pads and a system that senses the proximity of other robots, probably laser based but not clearly documented in the Kiva papers. With these sensors robots can avoid collisions. They also have a representation of their environment so they can decide how they move to a location of a product or station.

Collective Intelligence Collective Intelligent systems have properties that distinguish them from other systems. Collective Intelligence can be defined as groups of individuals doing things collectively that seem intelligent. To distinguish a collective intelligent system from another non collective intelligence system you can use many criteria. The AEGIRRRR [6] is the most extensive list. This list will be used to show the properties of a Collective Intelligent System.

1. adaptivity: an individual or the system adjusts itself to its environment
2. emergence: the whole is more than the sum of its parts
3. global-local: there is difference between the part level and system level
4. interaction: communication between individuals
5. rules: describing the behavior of the individuals
6. redundancy: knowledge is represented in a number of locations
7. robustness: lost representation cause by losing a part can be compensated
8. randomness: due to randomness small changes won't have any impact on the performance of the system

A system complying with these properties sounds promising. It should be the standard for robot based systems, but unfortunately it is not. Kiva almost complies with the definition used for collective intelligence. Kiva uses robots that react on each other and their environment. The robots perform product retrieval tasks, while the system as a whole process complete orders and ships them out to the costumer. Emergence is shown when traffic jams occur while a robot waits to be processed by a station, but also the placement of popular products near stations. Rules prevent collisions and provide order assignment. All robots have a representation of a part of the environment, while the whole system including the central server that sends the order messages has a representation of the complete environment.

Kiva only does not comply with the property randomness and also interaction is uncertain. Because the environment of a Kiva system will not be dramatically changed, except in emergencies like when a meteor crashes into the environment, randomness might not be needed. Implementing chaos could have a negative effect on the performance of the system, because it runs a lot smoother without. On the other hand smaller environment changing events, like blocked paths due to accidents, could occur and a more robust random system might be useful. If interaction is seen as communication between robots then Kiva lacks this property too. But if interaction is seen as avoiding and detecting collisions between robots it is a different story. If the central server that communicates with the agent is seen as an individual then Kiva also shows interaction.

Kiva does not have all the properties mentioned on the AEGIRRR list. This does not mean it is a bad quality system, because the properties of collective intelligence Kiva lacks, Kiva lacks with reason. It could also mean that there is still room for improvement.

1.3 Control

Kiva and other automated vehicle systems [1, 7, 5] had to make the choice if a centralized control method or a decentralized control method would be used. In a centralized controlled system all decisions are made by a central server. This central server has a complete overview, including robots, products and stations. This information is provided by stationary sensors. The system could reserve certain coordinates at specific times so high speed movement was possible and collisions could be prevented. With a decentralized control method there is no central server. All decisions are made by the robots themselves with or without help of external stationary sensors. This kind of system promises to be robust. When one component has broken down, no information is lost and there will be little change in overall performance. Kiva and other automated vehicle systems have chosen to make a combination of the two approaches. In Kiva a central server takes care of the resource allocation and the robots do everything else, like path finding, motion planning and task planning. When the central server malfunctions, it affects overall performance. The next section contains a decentralized approach for Kiva and automated system that should be more robust and also complies with the definition for collective intelligence.

2 Improvements of Kiva

Adjustment made	Improvement	Probability of implementation	Section
removing central agent selection	more robust system	low: companies prefer order over chaos so usually choose for a central controlled system	2.1
changing the product	adapting the product rack and the robot to another type of product like for pallet usage	high: Kiva can now be used in other environments like a production facility	2.2
replacing barcodes with RFID tags	less grid maintenance	medium: solves a current problem but the price level of RFID readers might need to drop	2.3
introducing cooperation	enables the system to handle jobs that can not be performed by a individual	medium:only using one type of product rack may be cheaper	2.4
introducing priority zones	speeds up retrieval process	high: no extra hardware costs, technique is used already in a different form	2.5
introducing area roles	prevents traffic jamms	high: no extra hardware costs and could boost preformance	2.6
introducing pheromones	alternative decentralized approach for first in first out systems	low: can only be used effective first in, first out systems	2.7

Table 1: Overview of Kiva improvements and changes

2.1 Decentralized alternative control method

This section gives an example of how Kiva system could be controlled so it would be more robust.

Overview All robots start near the processing stations. The only representation that is present at this time within the robots is the layout of the warehouse and the location of the processing stations and other stationary objects. When a order arrives the central server divides the order in multiple assignments and sends them to all robots. Because this server does not decide anything it could be seen more as a sensor than as an individual or as an agent. When received the first order, the robots start to move away from to stations. As soon as one robot finds the product from the assignment it communicates its success to all other robots. All other robots then stop searching for this particular product. The robot then moves the product to the process station that is described in the assignment. After processing, the robot stores the product. It searches for the nearest free storing space it can find and communicates the new location to all other robots. When a new assignment is broadcast the robots already carrying products may respond first. This assignment is then added to the assignment queue of the robot and when the robot is done with delivering the first assignment it will deliver the product to the delivery station of the intervening assignment before storing the product. For avoiding performance issues, robots can be given maximum amount of assignments robots can accept.

In depth: Selection No robots are selected to execute an assignment by a central server. All robots get the same assignment. When trying to find a product robots tend to spread out. This happens because a penalty function for staying together was implemented. When a product is found and a robot is selected to perform the assignment, this robot gets priority over all other non product carrying robots. If there are no assignments left, robots move of the paths and position themselves underneath a product rack or nearby charger. While a robot is not carrying out assignments it tends to stay out of the path of the product carrying robots. Robots can search for multiple products at the same time. The assignments send by the server are stored in the memory of the robot.

In depth: Movement When the product is found the robot receives more assignment information to which station it should deliver the product. The robot will take the best combination of main routes and product paths to deliver the product as fast as possible. Because this robot is carrying a product all other robots will move out of its path. It communicates its direction and speed to other nearby robots, but it could also communicate the reservation of some specific grid locations. Because robots know the locations of nearby robots, collisions are prevented. This information is given by the sensors the robot is equipped with which could be RFID, pressure or laser based. A combination of these three techniques could also be possible. When standing in a line, laser based sensors can be adjusted and touch based sensors can be activated to enable robots to be in a small area without causing any damage. Waiting in line to be processed by a station is one of the situation that occurs regularly and by minimizing the space between the waiting robots more space is available for storage. The earlier mentioned penalty system should be deactivated in such waiting areas.

In depth: Representation All robots have a detailed representation of the environment they are in. This representation is not complete, because it only contains the products that were used at least once and the products that have never been requested are still missing. When a product is unidentified for a certain amount of time, the product can be identified and moved to a location further away from the stations. Eventually each agent that will have a complete representation of the environment.

System Classification This system does have all the properties of collective intelligence and multi agent systems. It implements chaos and communication without much negative impact on performance. This approach also should be more robust than the original Kiva system, because it simplified the central server it uses to a sensor like level. Replacement of this central server would take less time than in the original system.

Communication Two main questions about communication within Kiva and similar systems are: “Is bandwidth a problem” and “How to handle fast roaming issues” [9]. Messages only have to be a few bits long. Even wireless networking products on the consumer market can offer up to 1 gig of bandwidth, so bandwidth should not be a problem. Fast roaming means switching between different access points when a robot moves out of the range of the access point it has a connection with. Normally this could take up to 30 seconds but with new roaming technology it could take 0.050 seconds. The new IEEE 802.11r facilitates this performance. This standard was meant to be used for voice applications.

2.2 Changing the product

Kiva already has different kind of robots and different kind of product racks. The product racks are custom build for specific kind of products and the robots are adapted to those product racks. So if the product changes in form, the product racks and robots usually change too. So what properties are needed and what need to be changed, if Kiva would be implemented in a distribution center that delivers complete pallets as products and uses trailers as delivery locations? In this system no human worker will be in direct contact with the product.

Side notes Trailers are docked at docking stations. These docking stations make sure that the trailer is perfectly aligned with the distribution center. Products are piled up to a certain height so they could fit inside a certain range of trailers.

Fig. 2: trailer and truck docked at a docking station of a distribution center

Still, in order to pile up a maximum number of products to a certain height, the product rack and therefore the robot, need to limit their height as much as possible. The surface of each trailer differs and the robot should cope with that. The surface can range from the usage of smooth wooden planks to the usage of damaged crooked wooden floor tiles. Also the transition between the building and the trailer is not perfect. A small difference in height of less then 1 cm might exist.

Overall changes Robots need to be equipped with shock absorbers and bigger inflatable rubber wheels in order to cope with the different types of surface and bumps. Also movement rules used inside the trailer need to be added. Because the trailer is small and long, only two robots can drive next to each other. Also the current way of loading a trailer by placing a row of products one row at a time is not efficient. So before robots enter the trailer they form a group. This group has enough products that can be used to fill the entire trailer or just a part of it. Because pallets are placed on product racks, robots can use the space underneath the pallets to all drive out at the same time. When traditional forklifts where used in this fashion to place pallets inside trailers , the majority of forklifts would be boxed in.

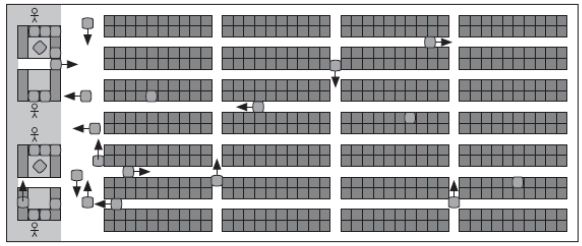
Movement In order to make movement inside the new trailer environment possible two solutions can be thought of. One solution uses a trailer equipped with a barcode grid and the other uses robots equipped with lasers to estimate the width and length of the trailer. The solution that uses the barcode grid sounds more plausible though it can only be used with custom trailers. Equipping trailers with this grid by placing printed barcodes can be done with almost no costs and when the distribution center has its own trailers, this job has to be done only once. Equipping robots with lasers is much more expensive even when only a part of the robot force will be equipped.

2.3 Barcodes

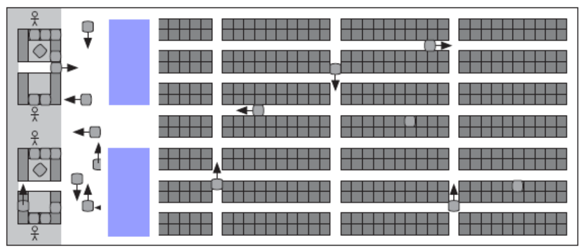
At this moment Kiva uses barcodes to identify grid locations. This is because it is a cheap and low-tech solution. Unreadable barcodes can be easily replaced by printing a new barcode-sticker and placing this new sticker at the unreadable location. Printing barcodes can be done for almost no cost at all. Dust, dirt and friction have a negative influence on readability of barcodes. Also multiple barcode lasers are needed to ensure a location is read. If only a single barcode laser is used, the robot has to be in perfect alignment with the barcode to read it. Remote Frequency Identification could be a good replacement for barcodes. For grid locations, cheap passive RFID tags can be used that are still a lot of more expensive than printed barcodes but they do not need the same level of maintenance as with barcodes. While covered with dirt and dust, a RFID tag can still be read by a reader. Even while sank in to the ground to avoid friction the readability of the tag won't change. RFID location sensing is possible [3], so when robots are equipped with active RFID tags that have a bigger range, robots now know the location of near robots without using communication. Another advantage is that the robot now can identify a product at greater distance. Many companies already implemented RFID in their current warehouse system. So when Kiva is introduced it should cost little effort to let Kiva use the RFID technology. But before RFID could be implemented for all Kiva systems at least one event should should take place: The price of active RFID readers should decline. Active RFID readers currently cost several hundreds of euro's so active RFID readers are just to expensive to be used on hundreds of robots.

2.4 Teamwork

Cooperation like seen in ant colonies, when an ant tries to move food that is to big or to heavy to be moved by one ant, a group of ants help. This behavior can also be introduced into Kiva. The idea of cooperation is not new to robotics [4, 6] but is new to Kiva. In the original Kiva system cooperation is limited in completing a customers order together. The kind of job where cooperation is needed to move a product with non normal size, is not an uncommon event in a distribution center. Sometimes bigger machine parts that are used for production facilities connected to the distribution centers need to be stored. To enable cooperation in a Kiva system some rules need to be added. For instance robots may only move when all the places underneath the product rack are filled and the robots that are placed underneath the product may only move together as a group. This can be done when one robot is assigned as a leader and the barcode coordinates it reads will be used for group movement. These kind of products can not be placed anywhere in the warehouse, because otherwise navigation and product placement would be impossible. Certain zones near stations or trailers might need to be reserved for placement and movement of these special sized products.



(a)



(b)

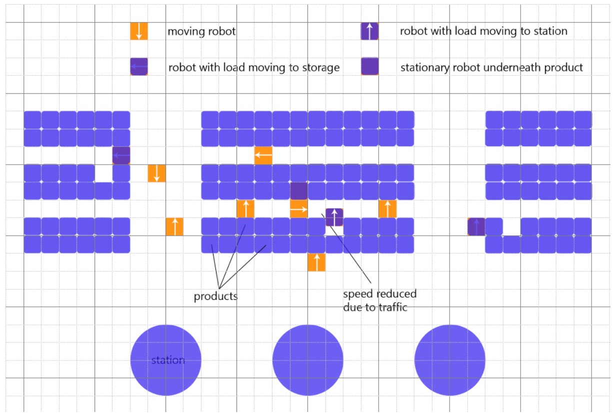
Fig. 3: a) normal Kiva [9] situation b) kiva situation with priority zones

2.5 Resource allocation: Priority Zones

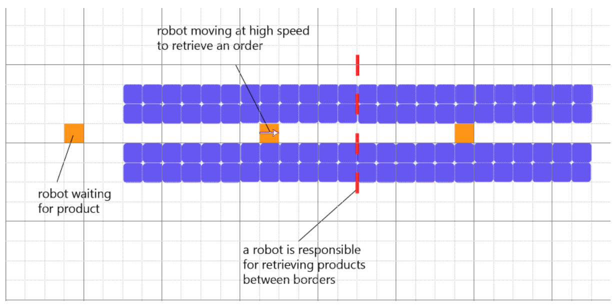
Instead of returning a used product to the storage area, a new area could be created between the storage area en station area. Products that are frequently used for lets say 10 times, within the last 5 minutes, can be stored in a priority zone. This saves search and retrieving time of products. If the product is not used for a certain amount of time, for lets say 3 hours, a special order can be given to relocate the product in question to the storage area. This is a different solution than used by the Kiva system, that relocates popular products as near the stations as possible, but only at free storage coordinates.

2.6 Specialization: Roles and Area

Specialization like with specific ant colonies where individuals specialize in a certain task, like working ant or nest defender, while still maintaining basic skills in both, could be a great idea when used with robotics in automated warehouses. One topic that is not mentioned yet in this paper is traffic and traffic jams. Hundreds of robots in small product paths can lead to traffic jams. Giving those robots specific roles might be a solution. A searcher robot, that searches for products in a product path and transports the requested product to the end of that path, could be implemented. The product is then picked up by the next robot, the transporter, which transports the product to a station.



(a)



(b)

Fig. 4: in a Kiva system there can be situation where the system is slowed down due to traffic. b) introducing specific roles bounded to specific areas could prevent these situations

This approach with only using 2 simple roles might improve performance. The traffic in narrow product paths is minimized so robots can move at greater speeds through product paths. Robots given a certain role do not have to be equipped with more hardware than mentioned in the decentralized system example. It is possible to leave out some hardware to save costs but when all robots are equal it is simple to let them switch roles. This might be needed when a specific robot has broken down.

2.7 Pheromones

Another concept from ant behavior might be an alternative for the usage of sensors to search for products. This concept is called pheromones. When foraging for food, ants leave a trace of pheromones on the route they take. The same happens when an ant has found food and transports it to the nest. In this approach the system grid is equipped with RFID tags instead of barcodes. All robots are equipped with low range RFID readers. The RFID readers can write short messages on these RDIF tags. When a robot transports a product over a tag, the reader stores which product is transported. This approach works like breadcrumbs, only these breadcrumbs can be uniquely identified so each product can be found. This approach could be used for a first in first out system, where a robot uses the first product of its kind to fulfill an assignment.

3 Conclusion

While the Kiva system is successfully implemented in warehouses and the system looks like a great alternative for automated vehicles and conveyor belt systems, there is still much room for improvement. Changes like replacing barcodes with RFID tags, implementing teamwork, introducing priority zones and roles might happen soon, because they are easy to accept and relatively cheap changes that could boost performance. The results of implementing decentralized control or usage of pheromones with Kiva might never be known because warehouse management usually prefers order above chaos and without these investors there will not be any budget to test these changes. It makes sense that in the near future Kiva will also develop pallet product solutions, because the production facilities that could use such a system are in direct contact with current Kiva warehouse clients.

References

1. R. Alami, S. Fleury, M. Herrb, F. Ingrand, and S. Qutub. Operating a large fleet of mobile robots using the plan-merging paradigm. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 3, pages 2312–2317, 1997.
2. R. D'Andrea and P. Wurman. Future challenges of coordinating hundreds of autonomous vehicles in distribution facilities. In *International IEEE Conference on Technologies for Practical Robot Applications*, TePRA 2008, pages 80–83, November 2008.
3. Marcel Flipse. Rfid techniques for indoor warehouse location sensing. http://marcel.flipse.itwillwork.nl, 2011.
4. C. Ronald Kube and Eric Bonabeau. Cooperative transport by ants and robots. *Robotics and Autonomous Systems*, 30(Issues 1-2):85–101, January 2000.
5. C. Le Pape. A combination of centralized and distributed methods for multi-agent planning and scheduling. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 488–493, May 1990.
6. Martijn C. Schut. Scientific handbook for simulation of collective intelligence. online, http://www.cs.vu.nl/ schut/dbldot/collectivae/sci/sci.pdf, 2007.
7. Danny Weyns, Kurt Schelfthout, Tom Holvoet, and Tom Lefever. Decentralized control of e'gv transportation systems. In *Proceedings of the fourth international joint conference on Autonomous agents and multiagent*, AAMASS '05, pages 67–74, 2005.

8. Michael Wooldridge and Nicholas R. Jennings. Intelligent agents: theory and practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.
9. Peter R. Wurman, Raffaello D'Andrea, and Mick Mountz. Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI Magazine*, 29(1), 2008.