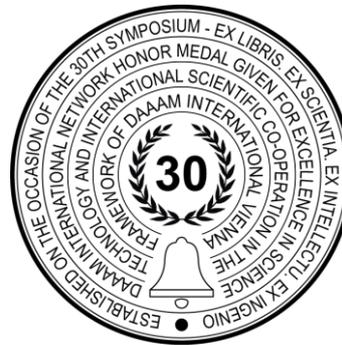


# A ROS TOOL FOR OPTIMAL ROUTING IN INTRALOGISTICS

Xenofon Karamanos\*, Panagiotis Mallioris, Dimitrios Poulimenos,  
Dimitrios Bechtsis & Dimitrios Vlachos



**This Publication has to be referred as:** Karamanos, X[enofon]; Mallioris, P[anagiotis]; Poulimenos, D[imitrios]; Bechtsis, D[imitrios] & Vlachos, D[imitrios] (2019). A ROS Tool for Optimal Routing in Intralogistics, Proceedings of the 30th DAAAM International Symposium, pp.0832-0838, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-22-8, ISSN 1726-9679, Vienna, Austria  
DOI: 10.2507/30th.daaam.proceedings.115

## Abstract

Robot Operating System (ROS) is a meta-operating system with a collection of sophisticated tools that can be used for enabling the scheduling and planning of mobile robots at industrial environments. Highly skilled IT professionals are handling intralogistics activities by programming the robots' behaviour at the facility layout, both at simulation and real-world environments. The customization of the system by non-experts is not a trivial task.

We propose an efficient and user friendly ROS tool as an interface for providing hot-spots at the facility layout where the loading and unloading activities take place. The user input is stored and visualized while an Ant Colony optimization algorithm determines the shortest path for navigating at the facility layout while passing through all the sections of interest. The final output is visualized at the simulation environment while in the meantime the real world robot follows the proposed path at the facility. Our pilot case study uses the Turtlebot3 Waffle mobile robot at a warehouse facility layout and the results indicate that the proposed tool could be utilized by non-experts for efficiently customizing the mobile robot's path in real-world conditions.

**Keywords:** ROS tool; Ant Colony Optimization; Planning; Mobile Robots; Intralogistics.

## 1. Introduction

The manufacturing industry after passing from the mechanization to the automation and the mass production phase is now to the challenging Industry 4.0 era. The Industry 4.0 revolution has led to automated and flexible industrial environments where repeated, time consuming and in many cases critical tasks are taken over by robotic systems. Heavy duty everyday activities are no longer considered a burden for human workers as they can be safely and reliably assigned to automated machines. Flexibility is a major characteristic of modern factories as the equipment can be easily customized in order to produce small lots of products and create resilient supply chains [1]. In this context mobility is an issue at the facility layout, as it could lead to intelligent systems that can increase productivity and decrease production down-times. Digital manufacturing motivates critical changes to the industrial landscape by the means of intelligent vehicles [2]. Either manually teleoperated, semi-automated or fully autonomous, mobile vehicles have been successfully used at production lines basically for transferring goods and materials (products, spare parts), human workers and even machines in the context of flexible manufacturing. Their penetration to the industrial facilities and to the supply chain ecosystem is

constantly increasing as they provide (i) 24/7 services, (ii) minimization of labour cost, (iii) low maintenance cost, (iv) enhanced accuracy in daily activities and (v) improved safety [3].

Automated Guided Vehicles (AGVs) are used in real-world scenarios at the manufacturing industry and logistics and provide intelligence, reconfigurability, flexibility and customizability to the industrial environment while relying less on human intervention [4]. Taking into account their utilization we can initially classify industrial vehicles according to (i) vehicle's type and size (chassis), (ii) the vehicle's kinematics (type of vehicle, type of wheels and movement) (iii) the navigation characteristics (embedded sensors, controls, facility layout) [3]. Moreover, AGVs' manufacturers provide sophisticated software tools for controlling vehicles at the production line. Software systems can have a centralized or decentralized architecture depending on the application and the field of use. Centralized systems are less flexible and usually require prior knowledge of the facility layout; on the other hand decentralized systems tend to enable the independent movement of each vehicle to the facility while the vehicles gradually become acquainted with the environment.

Depending on the vehicle's characteristics AGVs can navigate at the facility by localizing themselves and creating the occupancy maps for mapping the facility's layout. The occupancy map enables the vehicle to plan its movement from one point to the other creating a global path at the map while avoiding static obstacles. At the same time a local path is dynamically created in order to avoid moving obstacles that could intervene with the AGV's global path. Finally, in order to schedule and plan the AGV's complete trip in a facility, the authors focus on the well documented Vehicle Routing Problem (VRP) and consider the optimal route for visiting the regions of interest while minimizing the total distance travelled. The software optimizes the vehicle's route by minimizing an objective function (cost function) subject to a critical parameter like the total distance travelled by the vehicle. Experienced personnel (usually highly skilled technicians) use tools for programming the vehicle's behaviour at the facility. Major or minor changes at the facility must be carried out by authorized and special trained users in order to avoid collisions and inappropriate robot behaviour.

In this research paper, our focus is on providing sophisticated tools for utilizing the vehicles at the facility layout in a user friendly manner enabling the vehicle's control from non-experts. As a first step, the authors used the vehicle's sensors and Robot Operating System (ROS) for navigating and exploring the facility in order to create the facility's map. At a later step the proposed tool provides a visual interface for storing regions of interest (hot-spots) at the facility layout where the loading and unloading activities take place while the authors focus on routing optimization methods in intralogistics. An Ant Colony Optimization algorithm determines the shortest path for navigating at the facility layout while passing through all the sections of interest. The final output is visualized in the simulation environment while in the meantime the real world robot follows the proposed path at the facility. Our pilot case study uses the Turtlebot3 Waffle mobile robot and the results indicate that the proposed tool could be utilized by non-experts for efficiently customizing the mobile robot's path in real-world conditions.

The remainder of the manuscript is organized as follows. In Section 2 the Literature review sheds light to the context of incorporating AGVs in intralogistics, Material and Methods in Section 3 analyse the proposed software tool and in Section 4 we present the Results of our research work. Finally in Section 5 we conclude with the Conclusions and Next Steps work section.

## **2. Literature review**

Industry 4.0 is undoubtedly the main trend in manufacturing. Industries and Information Systems converge in order to provide an integrated environment for optimizing manufacturing processes and intralogistics. Automation is accompanied with business intelligence and factories are transformed for enabling intelligent manufacturing [5]. In order to establish an integrated system we have to focus both on hardware and software issues and promote interoperability at the shop floor. Machinery, individual sensors, mobile vehicles and devices become a unified ecosystem in order to communicate both with each other and with the human workers. Industrial Internet of Things (IIoT) handles the communication of the ecosystem and the 24/7 exchange of critical industrial data. On top of them Information Systems are parts of the ecosystem and conclude to decisions in order to support the human workers to operational, tactical and even strategic decisions. This technological innovation promotes profound changes to the way the stakeholders communicate with each other and promote Logistics 4.0 at the supply chain ecosystem [6]. Autonomous vehicles have the ability to act as mobile agents that live on the shop-floor, monitor the real-world infrastructure, act as a node to aggregate information from the shop-floor and even undertake certain actions according their actuators; from simply transporting goods and materials to actually undertaking critical manipulation of the machinery at the shop-floor using robotic arms. AGVs navigate at the shop-floor with levels of autonomy that start from simple teleoperated systems in known environments, proceed with the use of sensors in order to demonstrate basic behaviours, move on to mid and high complexity environments by gradually decreasing the human interaction levels and finally approach the fully autonomous intelligent characteristics in extreme environments at the highest complexity missions [7]. From another point of view, we should consider the six levels of autonomy scale in autonomous driving, that face the transition from human to autonomous driving beginning at Level 0 from no automation and gradually adding automation until the Level 5 where full-time autonomous features are met. At the final level perception and control reach similar levels (if not better) with the human perception levels in unknown environments [8]. As it is clearly stated the trend is the use of autonomous vehicles that could navigate at the industrial facilities without human intervention and conclude to decisions in order to increase productivity.

As stated by [3] the AGVs ecosystem consists of three major domains the surrounding environment, the hardware and the software used. The environment must be carefully considered in order to select the AGVs that best fit to the facility layout and can carry out the demanding daily activities, while enabling the use of certain technologies for the vehicle's navigation. The Hardware type includes the vehicle's type (forklift, tow, clamp, custom vehicle), the type of the mechanical parts (frame, motors, units on top), the electronic devices and the power source and finally the software type that includes the centralized, decentralized or hierarchical structure of the system and the algorithms used for the planning, scheduling, dispatching, routing and collision avoidance techniques. The presented research paper focuses on indoor intralogistics environments where material handling vehicles can autonomously navigate while avoiding obstacles. For indoor industrial environments and warehouses, Light Detection And Ranging (LiDAR) sensors and depth cameras along with inertial and odometry sensors are used for localizing the robots while secondary emergency systems assure the collision free movement of the vehicle at the industrial facility. Software tools can either have centralized control over the system (vehicles and resources) or decentralized control in case every node of the system could act independently. The use of a software systems for controlling AGVs at the shop floor is described at [9]. Path planning and scheduling is of major importance in software systems in order to provide minimum collisions on the shop floor and optimize productivity. The VRP is an NP-hard problem that considers the optimal path for visiting all the sections of interest while minimizing the total distance. Local search methods for solving the VRP for simultaneous pick-up and delivery activities focusing on a logistics environment are analysed at [10], while [11] simulated the pick-up and delivery of goods in a warehouse and optimized the AGVs' schedule. Finally, [12] elaborated on Simulated-Annealing with an Ant Colony implementation. The literature review section provided an insight to the work of other researchers and motivated us to develop the proposed software tool. Our scope is to create an open source software tool with a user-friendly Graphical User Interface (GUI) for non-experts in order to optimize intralogistics activities using AGVs.

### 3. Materials and Methods

As a framework for the creation of the proposed tool the authors used ROS. ROS provides a variety of sophisticated software tools for enabling robot communication methods and activities namely localization, mapping, path planning and scheduling methods and paves the way for the rapid reconfiguration of mobile robots at industrial environments [13]. For the actual movement of a vehicle in an industrial facility layout and for the simulation of the robot's movement at the map of the facility the authors used the TurtleBot Waffle kit that consists of a mobile chassis, 360 Laser Distance Sensor LDS-01, Raspberry Pi Camera Module v2.1, Raspberry Pi 3 Model B and B+, 32-bit ARM Cortex®-M7 with FPU (216 MHz, 462 DMIPS), RC-100B + BT-410 Set (Bluetooth 4, BLE), laptop computer and the additional TurtleBot mounting hardware kit. The mobile vehicle chassis was used in order to create the Occupancy Grid Map (OGM) of a real-world facility that clearly shows the boundaries of the facility, the static obstacles and indicates the free space for the vehicle's movement. At a later step the Graphical User Interface (GUI) was designed and implemented on top of the Rviz tool. Non-expert users pinpoint the regions of interest at the facility and the vehicle navigates through them using an optimization algorithm for minimizing the total distance travelled. The proposed software tool is demonstrated both at the simulation environment and at the real-world facility layout.

#### 3.1. Creation of OGM with Turtlebot and Slam

The initial step before navigating into an unknown environment is to create a perception of the environment. At our case study we start with the creation of the facility's map using the Turtlebot Waffle mobile robot. The robot is teleoperated at the facility and creates the OGM using the Simultaneous Localization and Mapping (SLAM) method. Given a series of controls and sensor observations over discrete time steps, the SLAM problem is to compute an estimate of the agent's location and a map of the environment. In our case study, the mobile robot estimates its current location with the help of the 360° Laser Distance Sensor. ROS supports various SLAM methods such as Gmapping, Cartographer, Hector, and Karto and in our research work we used the Gmapping method. The final OGM is presented in Fig. 1. Pixels with white colour indicate the area where the mobile robot can freely move, black colour represents the occupied area with the static obstacles and finally grey pixels indicate the unexplored regions.



Fig 1. The OGM of the experimental facility layout

3.2. The proposed tool's Graphical User Interface (GUI)

Following the 2D contour creation, the map is uploaded at the Rviz simulation environment. Rviz provides a friendly GUI for monitoring the real time activities of the mobile robot and communicating with the environment and the robot. Rviz, is a 2D and 3D visualizer for displaying the vehicle's environment, sensor's data and state information from ROS. Using Rviz, we can visualize current configuration and state of the mobile robot and the 2D contour of the OGM. Furthermore, we are able to display real-time information from ROS Topics (camera data, LiDAR distance measurements etc) and even information about the optimal trajectories created from Global and Local Path planners. Finally with the addition of Qt we can create a user friendly interface inside Rviz's window frame enabling the customization of the system by non-experts. Qt is a free and open source software for creating cross-platform applications that run on various software and hardware platforms such as Linux, Windows, MacOS, Android or embedded systems and is used for the implementation of the proposed software tool. The authors' work focuses on creating a user friendly application. The functionalities of the GUI are summarized as follows:

- Add, Delete points of interest at the facility and calculate the from-to distance matrix of the points of interest (Fig. 2)
- Run the algorithm and present the optimized solution for navigating to all the points of interest with the minimum travelled distance (Fig. 3)
- Customize the parameters of the ACO optimization algorithm used for the minimization of the total travelled distance by the mobile vehicle (Fig. 4)
- Visualize the results.

As a first step the non-expert user opens the 2D OGM of the facility that was created with the SLAM method. The user selects the Add button to insert a specific point of interest in the facility and the coordinates of the point are presented at a text box. Multiple points of interest are inserted in the map and when finished, the NavFn global path planner is used to measure all the optimum distances between any two points and form the From-To matrix of the points. The Global planner is a class that adheres to the `nav_core :: BaseGlobalPlanner` interface specified in the `nav_core` Ros package. `Nav_core` is a package that provides common interfaces for specific robot actions such as `BaseGlobalPlanner`, `BaseLocalPlanner`, and `RecoveryBehavior` interfaces, which can be used to trigger specific actions. As a result, with the addition of C++ code, the authors identify the optimal distance costs from the Global planner and store them at the From-To distance matrix. Finally the ACO algorithm takes as an input the From-To distance matrix in order to provide the optimal route that the simulated robot model has to follow, in order to minimize the total travelled distance. The optimal route is also visualized at the GUI.



Fig 2. The user adds points of interest, the tool calculates and visualizes the from-to distances

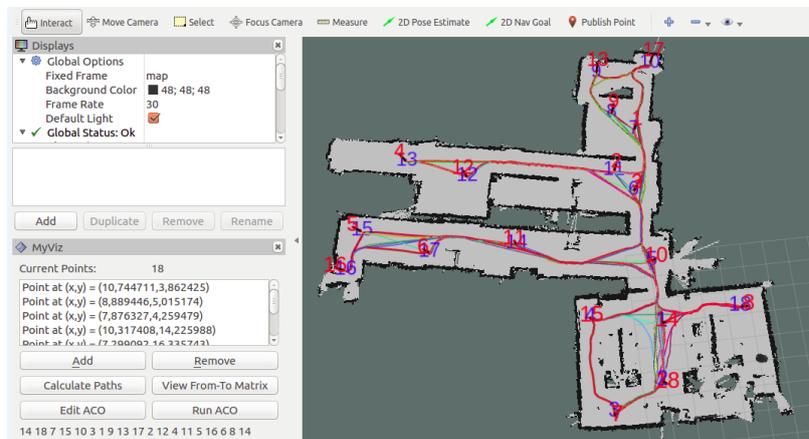


Fig 3. The final output of the tool, reorders the points and visualizes the minimum travel path (red line)

### 3.3. ACO implementation

Given a matrix of distances between the regions of interest in the occupancy map, the authors focused on identifying a closed Hamiltonian cycle that creates a path travelling through all the regions, and passing from any region once and only once. In order to find the optimal path we minimized the total traveling distance. This is exactly the definition of the world-famous Traveling Salesman Problem (TSP), a simplified version of the VRP. TSP is a NP-complete problem which means that finding an analytical solution is not an easy task and thus numerous algorithms have been developed throughout the years both analytical as well as heuristics and meta-heuristics [14]. For the optimal solution that minimizes the total travelled distance we implemented a ROS based variation of a meta-heuristic optimization algorithm, the hybrid ACO. A similar algorithm was implemented in a stand-alone C# tool that was used for warehouse intralogistics for solving the Capacitated Vehicle Routing Problem with Multiple Trips [15]. Just as the name suggests this algorithm at its basis mimics the behaviour of ants when trying to find the optimal path from their nest to their food source and back while reposing a special hormone that is called pheromone. On top of this algorithm, the authors included various optimization algorithms and techniques to overcome some important weaknesses of ACO such as its ability to get trapped to locally optimal solutions. Since this hybrid algorithm was developed for solving the Capacitated Vehicle Routing Problem with Multiple Trips which is a lot more complex than the Traveling Salesman Problem, it was decided that the algorithm would be modified, by removing unnecessary improving techniques such as the 2-opt algorithm and the swapping and insertion technique, thus improving its speed while maintaining accuracy. The algorithm was implemented using the C++ programming language in order to be used as a ROS plugin and its accuracy was tested in various benchmarks.

When running the software, the user provides and stores at the developed tool the parameters of the ACO algorithm in order to override the default values (Fig. 4). To begin with the number of Iterations must be defined. The default values are between 1000-3000 depending on the number of the nodes in the facility. The number of ants is represented with  $m$ , with recommended values of 10-20, and the probability  $q_0$  which determines the exploration ability of the ants. Lastly the parameters in the transition probability  $b$ ,  $x$  and  $a$  are defined with recommended values 2, 0.1 and 1 respectively, followed by the exhaust rate of the pheromones  $r$  with recommended value of 0.1. The values of the parameters are critical and define the accuracy of the algorithm and the user should always be careful when declaring them.

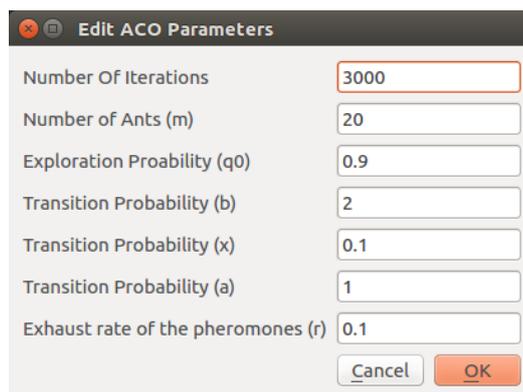


Fig 4. ACO customization interface

## 4. Results

The proposed tool provides a complete solution for providing the a set of regions of interest in a facility layout while the implemented ACO algorithm optimizes the tour through the set of points of interest so that it minimizes the total travelled distance. The validity of our solution has been initially tested with a set of benchmarks like the Western Sahara (29 points) and the Djibouti (38 points) with fair results (error percentage lower than 0.05%). As a next step we elaborated with our tool in order to run tests within the facility layout using different entry points. The Results table (Table 1) presents the shortest path at each case while Fig. 5 indicatively presents the final results (vehicle's global path) of each case study (10, 20 and 30 points).

Number of points at the facility	Total Distance (m)	ACO Computational Time (sec)
10	101.43465	8.378
20	113.78147	21.806
30	126.32276	42.394

Table 1. Results of our proposed tool

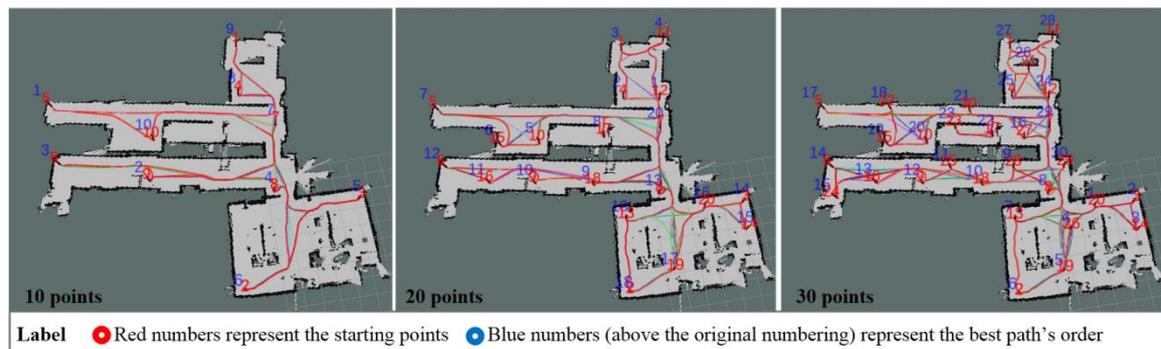


Fig 5. Three case studies using 10, 20 and 30 points at the facility and the automatically generated shortest path

Finally the authors elaborated with the Turtlebot robotic vehicle for navigating at the real-world facility layout using the output of the proposed software tool. The list of the points that form the best path was sequentially propagated to the vehicle and as a result it followed the predefined route.

## 5. Limitations and Conclusions

The OGM of our research presents the indoor environment of the Logistics and Supply Chain Management laboratory and this sets some limitations regarding the environment of the mobile vehicle. At a typical industrial layout the mobile vehicle confronts multiple dynamic obstacles, it uses the local planners to change the programmed route and this results to longer paths. Moreover we optimized the total travelling distance without taking into consideration the vehicle's capacity and any required time windows for picking-up and delivering the products.

In modern warehouses picking activities should be automatically executed in order to optimize space utilization and minimize the pick-up and delivery times. At the proposed research work a software tool was developed on top of the ROS environment for optimal routing in intralogistics. ROS basic activities were integrated in our tool and a custom algorithm was implemented for optimizing the total travelled distance. Our intention is to focus on non-expert users and support their daily activities by enabling the reprogramming of high-tech equipment including autonomous vehicles through custom build interfaces. Non-expert users add specific regions of interest in a given facility layout and the autonomous vehicle selects the shortest route for travelling through the set of regions. This is critical for optimizing the daily activities of an industrial layout (industries, warehouses) as we could reduce energy consumption and save time without the need of highly skilled personnel. Finally the tool provides interfaces for visualizing the vehicle's movement at the facility.

The proposed tool could be used in conjunction with the company's Warehouse Management System (WMS) system in order to prepare the daily orders. The WMS could analyse the orders, create a list of the corresponding products and indicate the exact location of every product at the facility. Products could be collected by the vehicle while our proposed tool will handle the vehicle's navigation, path planning and scheduling. For the picking activities the vehicle could either use actuators (eg robotic arms) or simply pass through the shelves and stop at the predefined locations while human workers load the products. Safety regulations are currently at an infancy state [16] and the authors plan to further study safety concepts and include them at a future version of the software tool.

## 6. Acknowledgements:

This paper was partially supported by the GSRT project "Dynamic Quality CONTROL on Production lines using intelligent Autonomous vehicles (Q-CONPASS)"

## 7. References

- [1] Bechtsis, D., Tsolakis, N., Vlachos, D. & Iakovou, E. (2017). Sustainable supply chain management in the digitalisation era: The impact of Automated Guided Vehicles, *Journal of Cleaner Production*, Vol. 142, pp. 3970-3984.
- [2] Scarinci, R.; Hegyi, A. & Heydecker, B. (2017). Definition of a merging assistant strategy using intelligent vehicles, *Transportation Research Part C: Emerging Technologies*, Vol. 82, pp. 161-179.
- [3] Bechtsis, D.; Tsolakis, N.; Vouzas, M. & Vlachos, D. (2017). Industry 4.0: sustainable material handling processes in industrial environments, *Computer Aided Chemical Engineering*, Vol. 40, pp. 2281-2286
- [4] Mehami, J.; Nawi, M. & Zhong, R. Y. (2018). Smart automated guided vehicles for manufacturing in the context of Industry 4.0, *Procedia Manufacturing*. Elsevier B.V., 26, pp. 1077-1086. doi: 10.1016/j.promfg.2018.07.144.
- [5] Zhong, R. Y.; Xu, X.; Klotz, E. & Newman, S. T. (2017). Intelligent Manufacturing in the Context of Industry 4.0: A Review, *Engineering*, Vol. 3, pp. 616-630.

- [6] Barreto, L.; Amaral, A. & Pereira, T. (2017). Industry 4.0 implications in logistics: an overview, *Procedia Manufacturing*. Elsevier B.V., Vol. 13, pp. 1245–1252. doi: 10.1016/j.promfg.2017.09.045.
- [7] Huang, H.-M.; Albus, J. S.; Messina, E. R.; Wade, R. L.; English, R. W. (2004). Specifying Autonomy Levels for Unmanned Systems: Interim Report, SPIE Defense and Security Symposium 2004, Conference 5422, Orlando, Florida, 5422, pp. 386–397. doi: 10.1117/12.552074.
- [8] Zhang, T.; Li, Q.; Zhang, C.; Liang, H.; Li, P.; Wang, T.; Li, S.; Zhu, Y. & Wu, C. (2017). Current trends in the development of intelligent unmanned autonomous systems, *Frontiers of Information Technology & Electronic Engineering*, Vol. 18(1), pp. 68–85.
- [9] Eilers, K. & Rossmann, J. (2014). Modeling an AGV based facility logistics system to measure and visualize performance availability in a VR environment. In: *Proceedings of the Winter Simulation Conference 2014*. IEEE, Savannah, GA, pp. 367-375.
- [10] Gajpal, Y. & Abad, P. (2009). An Ant Colony System (ACS) for vehicle routing problem with simultaneous delivery and pickup, *Computers & Operations Research*, Vol. 36, pp. 3215-3223.
- [11] De Santis, R.; Montanari, R.; Vignali, G. & Bottani, E. (2017). An adapted ant colony optimization algorithm for the minimization of the travel distance of pickers in manual warehouses, *European Journal of Operational Research* (2017), <https://doi.org/10.1016/j.ejor.2017.11.017>
- [12] Skinderowicz, R. (2017). An improved Ant Colony System for the Sequential Ordering Problem, *Computers & Operations Research*, Vol. 86, pp. 1-17.
- [13] Andreev, V.; Kim, V. & Pletenev, P. (2017). The principle of full functionality – the basis for rapid reconfiguration in heterogeneous modular mobile robots // *Proceedings of the 28th DAAAM International Symposium*, pp. 0023-0028, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-11-2, ISSN 1726-9679, Vienna, Austria. DOI:10.2507/28th.daaam.proceedings.003.
- [14] Bollobas B. (1979). *Graph Theory: An Introductory Course*, New York, pp. 12.
- [15] Poulimenos, D, Bechtsis, D., Vlachos, D., Iakovou, E., (2018). A software tool for optimizing intra-logistic activities, *Computer Aided Chemical Engineering*, Elsevier, In *Computer Aided Chemical Engineering*, Elsevier, Volume 43, pp 439-444.
- [16] Papa, M; Kaselautzke, D; Stuja, K. & Wolfel, W (2018). Different Safety Certifiable Concepts for Mobile Robots in Industrial Environments, *Proceedings of the 29th DAAAM International Symposium*, pp.0791-0800, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-20-4, ISSN 1726-9679, Vienna, Austria DOI: 10.2507/29th.daaam.proceedings.115