Indoor Navigation System Using Dijkstra's Algorithm For Emergency Situation

Mohd Faizulhayad Bin Mashuri
Faculty of Electrical Engineering
Universiti Teknologi Mara (UiTM)
40450 Shah Alam, Selangor, Malaysia
faizul.mashuri16@gmail.com

***Abstract*— This paper will discuss on an implementation of route planning in Indoor Navigation System (INS) for emergency situation. Global Positioning System (GPS) cannot be used for INS since signals sent by satellites cannot penetrate the construction materials inside a building. Wireless Fidelity (Wi-Fi) is the system that can make INS possible. By implementing Radio Signal Strength (RSS) provided by Wireless Fidelity (Wi-Fi) where most building already have it, Reference Points (RP)s with it designated RSS can be setup across the area desired. The connection or paths between each RP can be created to create the best route to the destination. The technique used for the route planning is Dijkstra's algorithm. The result of this technique shows higher accuracy in terms of shortest distance taken to the destination compared to other technique and the implementation of this technique are easy since users only have to manipulate the weight between each RPs according to obstacle that exist along the paths. Bigger obstacles will gives higher value of weight. Route planning using Dijkstra's algorithm can produce better accuracy and more reliable in indoor environment since the only factor taken into account is the weight for each path.**

***Keywords* —Indoor Navigation; Dijkstra's Algorithm; RSS-Based Indoor Navigation; Fingerprinting Technique: Map Matching Technique.**

I. INTRODUCTION

When discussing on navigation, most of us will think about Global Positioning System (GPS) where we locate our position and our destination by the help of the satellite. Unfortunately, GPS signal cannot be received in an indoor environment such as in buildings where the nature of the satellite signal which is straight line of line signal that being obstructed by the walls and roof of a building [1][2]. Due to this this constrain, the existing navigation system does not provide indoor navigation service. Google Maps as we see is one of the most reliable source of navigation, where Google Maps provide detailed information about any maps that we searched but one of its limitation is Google Maps cannot display the space structure and and specific information inside of the building despite all the technologies they used to generate real-time maps display even in 3 dimension (3D). These limitation has inspired people to develop an indoor navigation system with high accuracy and give more detailed information about indoor environment such as in a building [2][3].

A navigation service has to be consistent in providing navigation for indoors and outdoor. A consistent connection between the provider such as satellite to the user can provide accurate information about the destination in terms of time of arrival, accurate place of destination, shorter route to the intended places and many more. So far, outdoor navigation applications are successful in providing consistent navigation to users, but indoor navigation are still improving in this matter [8]. By improving the indoor navigation system, users can explore and go the intended room in an unknown building without having the difficulties to ask strangers and if we look further, one of the advantages in using indoor navigation system is for emergency cases such as fire and earthquake where the system will provides route to the nearest exit in the building to the assembly point.

In todays world, we can find wireless network everywhere we go. It is almost a must for a building to have wireless network installed and act as one of the building's facilities so that people can access the internet and uses the Wireless Fidelity (Wi-Fi) freely as long as they still in the premises. Utilizing Wi-Fi signals and Access Points (AP) is a tempting approach where the results was encouraging [10]. If outdoor navigation have satellite that provides information of the location to the users, for indoor, Access Point or the Wi-Fi router will act as the 'satellite' where we can manipulate the Radio Signal Strength (RSS) to give us the information such as providing the location of the user and the destination. The RSS-based system is a system that based on time consuming calibration phase to build a radio map of the environment, which used to find the relation between the RSS measurement and position [4]. All information that necessary to create an indoor navigation system can be obtain by manipulating the RSS value from the Aps.

Basically for indoor navigation, there are three main parts that need to be taken into account which are localization, navigation and the last part is Graphic User Interface (GUI)[7]. Localization and path-planning are two most important matters that have to be solved in order to achieve proper navigation. Feasible localization can be done by implementing route maps that contain enough information in terms of knowing the position of the user and computing feasible trajectories towards the intended destination [9]. For this project, fingerprinting technique, one of the technique in RSS-based localization is used by manipulating the RSS value and the algorithm involved is K-Nearest Neighbor (KNN). As for navigation, there are few algorithm that can be used such as A* algorithm and Dijkstra algorithm to find the shortest path or route to any destination intended. For GUI, Matlab is used as the medium to process our raw data so that we can display the map and the route in manner where users can understand.

In this research, the researcher is focused on Dijkstra algorithm since it is more reliable and more accurate compared to A* algorithm. Using this algorithm, the weight for each path needs to be assign manually based on the obstruction that exist within each path that already being created. For this research, the researcher try to create two situation where the first situation, there is no obstruction exist in the path or an ideal situation where same weight is assigned for each path disregarding the path loss and the propagation loss. The second situation is the real-time environment where obstacles that exist within each path are included by manipulating the weight so that the shortest path from the location where the user are standing can be generate. The medium used in order to generate the shortest path is Matlab where the Dijkstra algorithm generated using the .m file and the input of this algorithm will be fetch from the KNN search algorithm that we run earlier during the localization process.

II. METHODOLOGY

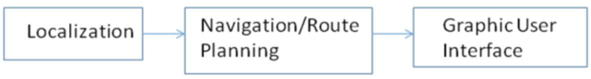

Figure 2.0 : Simple Block Diagram for Indoor Navigation System

This paper will focusing on the second part of the diagram which is the navigation part or the route planning part where we will be using Dijkstra algorithm to generate the shortest path from the starting location to the destination intended. In this case, the researcher are choosing emergency situation so the emergency exit is the intended destination.

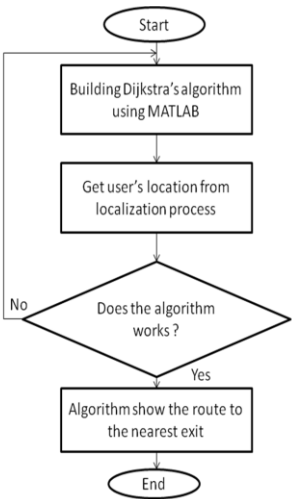

Figure 2.0 : Process Flowchart

From the flowchart, the process to build the algorithm were done by using MATLAB. All the coding and inputs were integrated in using this platform. For starter, all the weights was set to one (1) since the distance between each Reference Point (RP) were the same. Later some of the weights were changed to two (2) according to the obstacles that exist between some paths. Next step is the user's location that need to be initialize in the algorithm where the data are from localization process but in this research, the data for user's location were selected by the researcher manually, in this case the researcher choose point five (5) as the starting point. After the initial location of the user already being initialized, the algorithm will directly show the shortest distance based on weight and the route will be shown on the map.

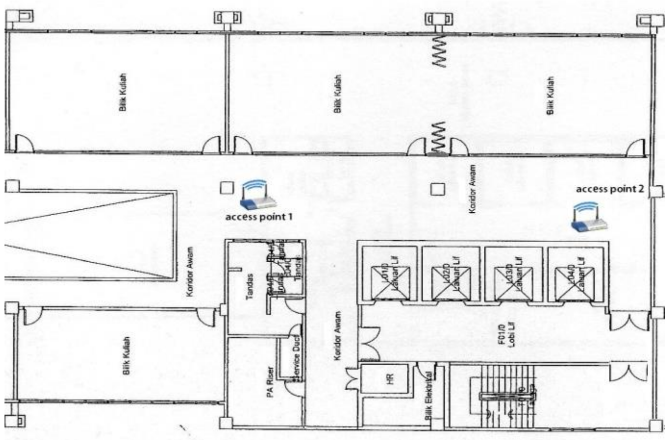

Figure 2.1 : Floorplan of the Block 5 Level 13 of Electronic Engineering Faculty of UiTM Shah Alam

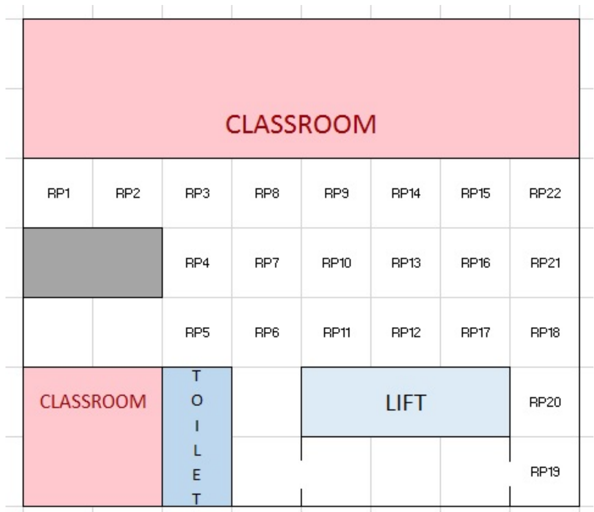

Figure 2.2 : Floorplan of the Block 5 Level 13 of Electronic Engineering Faculty of UiTM Shah Alam

From the Figure 2.2 , Reference Point 19 (RP19) is the intended destination which is the emergency exit of Level 13 Block 5 where the RPs are the node that already assigned manually and for each RP, it has its own unique coordinate generate from the values of the RSS taken during the fingerprinting process and all the values of the RSS are saved in th database that we created earlier.

The computation of the best route or the shortest path is based on Dijkstra's algorithm that widely used in the navigation system world where the algorithm can specifically estimate the shortest path between starting node and the destination node therefore will guide the user as they were to explore a floor in an unknown building [5].

Dijkstra's algorithm maintains an array of tentative distances , $D[u] \geq d(s, u)$ for every node that being assign. The algorithm will visit every nodes according to its distances from the starting node and maintains the invariant for the visited nodes, $D[u] = d(s, u)$. When a node, u is visited, its outgoing edges (u,v) are relaxed. Dijkstra's algorithm will terminate any nodes that already visited. As for the size of the search space, the average is $O(n)$ and $n/2$ [6]. Below are the general steps to apply Dijkstra's algorithm to solve the shortest path or to find the best route for most cases in navigation system.

Let the starting node to be called as the initial node. Let the distance of node *Y* be the distance from the initial node to *Y*. Dijkstra's algorithm will assign some initial distance values and will try to improve them step by step.

1. Firstly, assign tentative node for every nodes and as for the initial node, set it to zero and for the other nodes, set it to infinity.
2. Mark all nodes involved as unvisited. Then set the initial node as current. Create a set of the unvisited nodes called the *unvisited set* consisting of all the nodes.
3. For the current node, consider all of its unvisited neighbors and calculate their *tentative* distances. Compare the newly calculated *tentative* distance to the current assigned value and assign the smaller one. For example, if the current node *A* is marked with a distance of 6, and the edge connecting it with a neighbor *B* has length 2, then the distance to *B* (through *A*) will be 6 + 2 = 8. If B was previously marked with a distance greater than 8 then change it to 8. Otherwise, keep the current value.
4. When we are done considering all of the neighbors of the current node, mark the current node as visited and remove it from the *unvisited set*. A visited node will never be checked again.
5. If the destination node has been marked visited (when planning a route between two specific nodes) or if the smallest tentative distance among the nodes in the *unvisited set* is infinity (when planning a complete traversal; occurs when there is no connection between the initial node and remaining unvisited nodes), then stop. The algorithm has finished.
6. Select the unvisited node that is marked with the smallest tentative distance, and set it as the new "current node" then go back to step 3.

III. RESULTS AND DISCUSSIONS

As discussed earlier, two graphs will be generated based on two situation where the first situation is all the weight for each path taken will be assign with same value. In this case, the weight will set as 1for the first scenario. For second situation, the weight value for some paths will be changed and try to create real-time obstacles since there will always be obstacles in the paths in real world.

1) Weights = 1

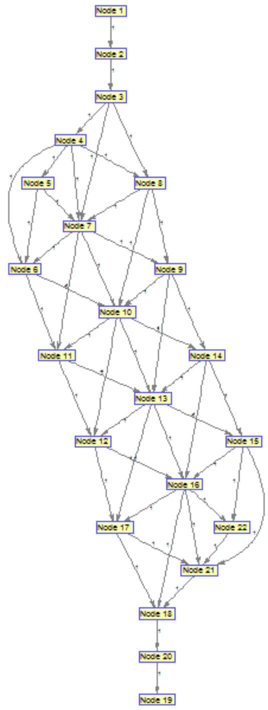

Figure 3.0: Dijkstra Graph with 22 Nodes and 52 Paths

From the graph above, all the weights have been set to 1 with the same number of nodes as in the floorplan in figure 2.2. Node's name in Figure 3.0 is equal to Reference Point (RP) in Figure 2.2. Although the location of the nodes are not the same as the one in the floorplan but the number of nodes are the same. The graph is just the representation of the Dijkstra's algorithm. From this graph, the initial idea on how the navigation will show the shortest route on the map or the floorplan can already being seen. By matching up the RPs name and the RSS value saved in the database, clear route will be shown on the map.

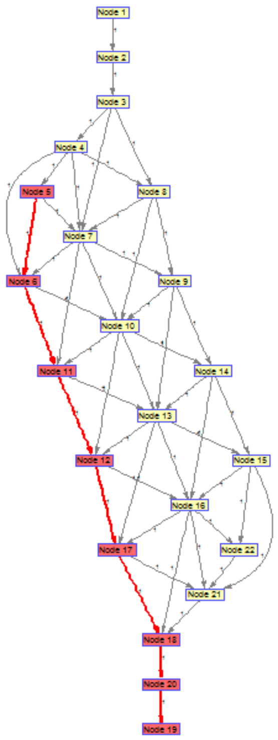

Figure 3.1: Shortest Route with the Initial Location to the Destination

The graph in Figure 3.1shows the shortest route from the initial location that being set according to the location of the user. In this case, the assumption is that the user is at node 5. Whe all the weights are set as 1, the shortest route to the destination will go through node 6, 11, 12, 17, 18, 20 and finally node 19. Although there are 22 nodes that already being assigned for the reference points, but the last node or destination node is not 22, instead node 19 will be the final node since the reseacher need to balance the node between the initial location and the destination in the algorithm. By using Dijkstra's algorithm, the shortest route to the destination can be quickly determine compared to other algorithm since the most important factor for Dijkstra's is the weight between 2 nodes. Next some changes will be shown to the graph if the obstacles are included where the weight for each path that have an obstacles such as study desk or dustbin will be changed, with the same initial location to the same destination which is the emergency exit or node 19.

2) Some weight are not equal to 1

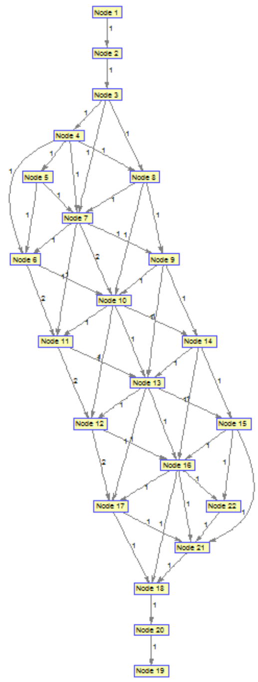

Figure 3.2 : Some of the weight has been change

For Figure 3.2, some of the weight has been change to 2 since in reality, there are study desks along the path from node 6 to node 17. There is also 1 study desk along the path of node 7 to node 10. So these routes cannot be access due to obstacles and since Dijkstra's algorithm will take the shortest route with reference of smaller weight, so the weight for these routes will be increased to 2. Higher than any other routes so that the algorithm will not choosing these routes during route planning.

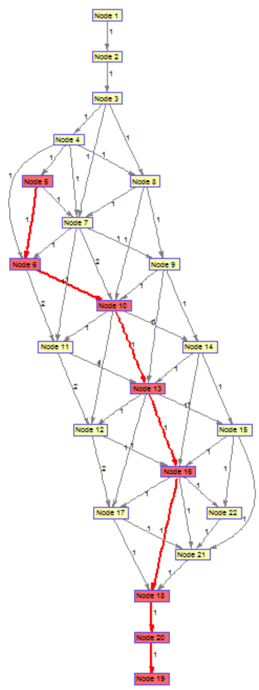

Figure 3.3 : New route plan after taking obstacles into account

As shown in the graph, after the same initial location used for the algorithm, the same node as in case 1, the route plan already changed. Instead of choosing route from node 5, 6, 11, 12, 17, 18 and 19, the algorithm has chosen route from node 5, 6, 10, 13, 16, 18, 20 and finally node 19 after taking obstacles into account. It can be said that if there are any new obstacles appear in the floorplan, by changing the weight of the route that having the obstacles, the algorithm will immediately gives the user the alternative route to the destination and still provides the user with the shortest path.

IV. CONCLUSION

As conclusion, route planning is an important part in navigation and one of the best technique is using Dijkstra's algorithm. One of the advantages for using this technique is the algorithm will directly show the user the route to the destination they desired based on weight only. By entering the the user location that we get from localization process, the map will directly show the user to the nearest exit in case of emergency such as fire or eatrhquake.

V. RECOMMENDATION

For route planning, Dijkstra's algorithm is the best algorithm to provide shortest path for route planning but to make it more reliable and accurate, this project can be upgraded by adding Inertial Measurement Unit (IMU) in the future where using these technique, the system can estimate the time of arrival and sensing the user's movement. Instead of only depending on the RSS value, other factors will be included such as the readings of gyroscope, magnetometer and accelerometer to improve the system's reliability and accuracy.

It is hoped that in the future, more destination can be added in a single navigation especially during emergency such as fire and earthquake so that users can directly go to the nearest exits around them, not depending only to one exit.

Other improvement that can be recommend is to design a backup system for these navigation system in case of anything happen to the Access Point (AP) where the system cannot update the user's current location thus route planning cannot be used, so this backup system will immediately provide the system with the last location of the user and provide the user with alternative route so that the user can still arrive to the exit point safely.

VI. ACKNOWLEDGEMENT

This project would not have been possible without the support of many people. First and foremost, I would like to express highest gratitude to my supervisor, Dr. Muhammad Fahmi Husin and my Co-supervisor, Dr. Idora Razak for the valuable guidance, advices and encouragement given since the beginning of my project until the end of my project.

I would also like to express my grattitude to parents who always give me encouragement and prayer especially during my hard and torrid time to complete these project. Without their prayer, I would not be able to complete this project on time.

Special thanks to all friends especially Felix Ceilomond anak Samam and Sharafuddin bin Khairudin due to the hardwork during the process of completing this project and never stop giving the motivations and encouragement.

VII. REFERENCES

[1] Wendlandt, Kai, M. Berhig, and Patrick Robertson. "Indoor localization with probability density functions based on bluetooth." *Personal, Indoor and Mobile Radio Communications, 2005. PIMRC 2005. IEEE 16th International Symposium on*. Vol. 3. IEEE, 2005.

[2] Seco-Granados, Gonzalo, et al. "Challenges in indoor global navigation satellite systems: Unveiling its core features in signal processing." *Signal Processing Magazine, IEEE* 29.2 (2012): 108-131.

[3] Hong-wei, Gao, Bi Xiang, and Q. I. Qin. "On positioning and navigation of mobile-phone based on gpsOne." *Microcomputer Application Technology* 69 (2007): 34-37.

[4] Tarrío, Paula, Juan A. Besada, and José R. Casar. "Fusion of RSS and inertial measurements for calibration-free indoor pedestrian tracking." *Information Fusion (FUSION), 2013 16th International Conference on*. IEEE, 2013.

[5] Gilliéron, Pierre-Yves, et al. "Indoor navigation performance analysis."*Proceedings of the 8th European Navigation Conference GNSS*. No. TOPO-CONF-2005-014. 2004.

[6] Dijkstra E.W, "A note on two problems in connexion with graphs : Numerische Mathematik 1", 269-271, 1959

[7] Jiang, Landu. *A WLAN Fingerprinting Based Indoor Localization Technique*. Diss. University of Nebraska, 2012.

[8] Asthana, Abhaya, Mark Cravatts, and Paul Krzyzanowski. "An indoor wireless system for personalized shopping assistance." *Mobile Computing Systems and Applications, 1994. Proceedings., Workshop on*. IEEE, 1994.

[9] Pacheco, L., et al. "Experimental results of an indoor navigation method based on bionic behaviors." *Automation Quality and Testing Robotics (AQTR), 2012 IEEE International Conference on*. IEEE, 2012.

[10] Kemppi, P., et al. "Hybrid positioning system combining angle-based localization, pedestrian dead reckoning and map filtering." *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*. IEEE, 2010.