IAENG_W

by Muhammad Rusli

Submission date: 04-Mar-2019 08:21AM (UTC+0700)
Submission ID: 1087029882
File name: IAENG_W.docx (532.54K)
Word count: 3769
Character count: 18948

Conflict-Based Multi-Capacity Constraint Route Planning

Abstract—Planning and scheduling mass departure is critical for tsunami disaster management. Algorithm that can generate effective evacuation plan and schedule is required. The evacuation plan and schedule have to ensure that the whole population can be moved to safety areas before the calamity of tsunami takes its impact on the land. Current heuristic solutions fail from computational **complexity by assuming that the network's elements, such** as nodes and edges, have infinite capacity. Their approaches generate lazy evacuation route and evacuation plan as one instance. They fail to ensure their scalability when dealing with a real problem, i.e. huge transportation networks. This paper introduces a refinement of a multi-capacity constrained heuristic routing algorithm, which embedded a conflict-based path generation for evacuation scheduling. The algorithm is exercised on tsunami scenario in a capital city (northern coast of North Sulawesi, Indonesia which has the population of more than 400,000 lives. The proposed algorithm improves the computation time significantly (80%) while maintaining the time required for executing the evacuation plan the previous solution

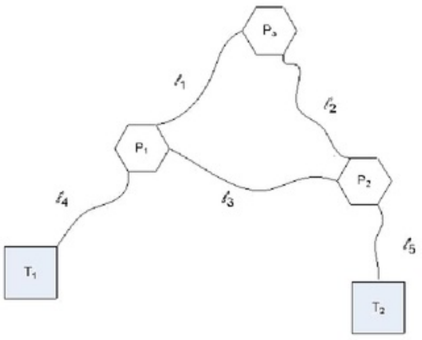
I INTRODUCTION

Planning evacuation route and schedule are a critical process for tsunami mitigation activities in a coastal area. An efficient and yet scalable computational algorithm is required to generate evacuation route and schedule that ensure zero life lost when the event of great tsunami actually takes its impact on the land. Some solutions focus on solving the problem as a linear programming problem [1, 2]. The computational complexity of the solutions is exponential, which are not suited for a huge network such as in urban area.

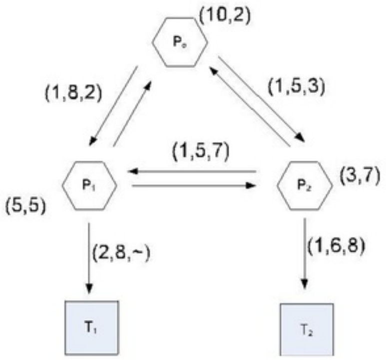
A number of heuristic solutions are proposed to solve the problem of exponential complexity of the computation [3, 4, 5]. The solution was inspired by the work of Goodchild et al. [6], which suggest the use of geospatial information for generating optimal evacuation routes. The solution proposed by Sarshar et al. utilizes the Volunteered Geographic Information (VGI) created by community [7]. This decision was taken due to the fact that of formal agencies (Environment System Research Institute) can only provide limited resources, which cannot match the rate of which the community producing the geospatial information [8]. Thus, the aforementioned solutions use a heuristic approach to generate optimal routes and or schedule. They reduce the computational time compared to the linear-programming approaches. These solutions generate the evacuation route with the assumption that any group of evacuees that transits at an evacuated node may be directed to take different evacuation path than other groups started from the evacuated node or also transits from that evacuated node but from different source node. They also assume that each group of evacuees from an evacuated node may be directed to take a different evacuation path than other group from the same node. These assumptions are not feasible when applied to the field. During the evacuation time, it is difficult to organize evacuees to join their designated group and to take their designated path. This is because the tsunami event may occur anytime and panic may disorient evacuees in choosing their designated evacuation path.

Sarshar et al. introduces a dynamic model based on Bayesian Network to predict the probability of congestion occurrences during evacuation process in ship [7]. The algorithm considers a number of variables, such as passengers' panic, passengers' gender, and the structure of the ship. It predicts the probability of trapping in congestion for evacuees and the extension of evacuation time given an evacuation model. Nonetheless, it does not consider the dynamic of the evacuation routes. Other research by Kinugasa and Nakatani focuses on providing a tool that analyzes the behavior of visitors, such as tourists, during an evacuation process [9]. The work compares and evaluates the effectiveness of various evacuation guidance methods. The case for the study is Kyoto city, especially the tourism area. It focuses on the event of an earth quake. It uses a static evacuation plan, where the state (availability) evacuation paths are considered unchanged.

The research by Ogunlana and Sharma evaluates an evacuation model which is based on intelligent agent [10]. It develops a visualization tool for that purpose. The tool helps in envisaging the time of evacuation. It also helps analyzing evacuation model based on the what-if scenario. It incorporates data on human emotions and movements.



(a) Evacuation Scenario



(b) Evacuation Graph

Fig. 1. Evacuation Problem Model

An enhanced algorithm based on Multi-Capacity Constrained Route Planning was developed. The algorithm is called Conflict-Based Multi-Capacity Constrained Route Planning (CBMCCRP). The algorithm separates the process of generating the evacuation route and the evacuation schedule and delays as long as possible the evacuation schedule. It adopts the lexicographic function to minimize the number of casualties. This algorithm was tested on real-scale tsunami case in Manado, which lies on the coast of North Sulawesi. It has to deal with evacuating more than 400.000 lives on more than 400 nodes and edges.

The rest of the paper is organized as follow. The next section paper describes how the problem of tsunami evacuation is modeled. Then, it describes the proposed algorithm called Conflict-Based Multi Capacity Constrained Route Planning. The paper then presents the experimental result and its analysis. The last section provides the conclusion of the paper.

II. TSUNAMI EVACUATION PROBLEM

Evacuation route planning can be best described as a graph as seen in Fig. 1. The figure in Fig. 1.a shows an evacuation scenario with 3 evacuated node (P_0, P_1, and P_2) and two safe nodes (T_1 and T_2). In this scenario, the population in each evacuated node has to be evacuated in given time. In this example, all evacuees in P_0, P_1, and P_2 have to be evacuated within 2 hours, 5 hours, and 7 hours respectively after the first warning given. The edges (roads) l_1, l_2, and l_3 connect two nodes and may be unavailable within different time after the first warning. For example, after 2 hours, l_1 will be disconnected because tsunami takes its impact on that edge.

This evacuation scenario can be represented as a *graph* $G = (N = P \cup T; A)$, *where* P *and* T *are the set of evacuated node (and/ transit node) and safe node respectively, and* A *is a set of edges* l. *Each evacuated node* i *has a set of attributes* (d_i, f_i), *which consist of a*

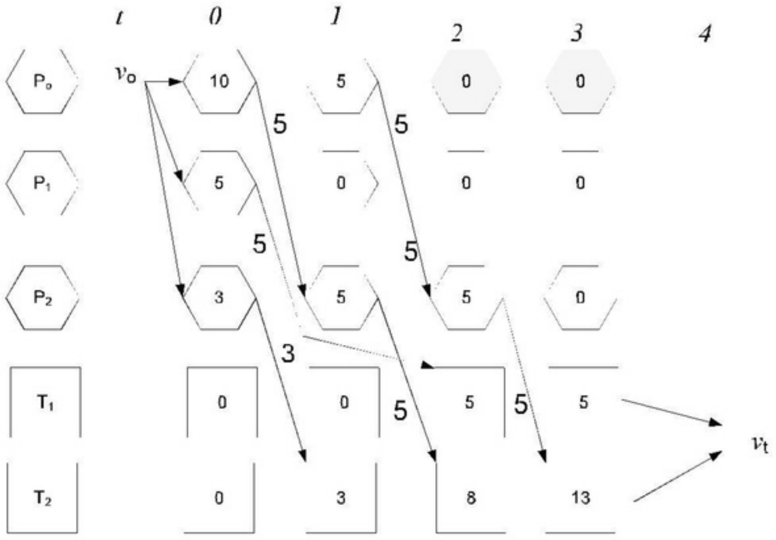


Fig. 2 Time-Expanded Graph of an Evacuation Scenario

number of evacuees and the deadline time for evacuation respectively. Each edge l has a set of attributes (s_e, u_e, f_e), which includes distance of the edge (with respect to time needed for evacuees to travel from source node to destination node), edge capacity (a maximum number of evacuees that can be in the edge at an evacuation time unit), and the time when the edge become unavailable respectively.

III. CONFLICT-BASED MULTI-CAPACITY CONSTRAINT ROUTE PLANNING

Since the CBMCCRP adopts the conflict-based approach in generating the evacuation path of a specific node, CBMCCRP models the evacuation problem as a time-expanded graph by discredited the planning in time steps with identical length. Fig. 2 illustrates the time-expanded graph of the previous scenario. Graph $G^d = (N^d = P^d \cup T^d, A^d)$ is generated by duplicating each node in *N* and edge in *A* for each time *t*. For each time *t*, any unavailable node in N^d and edge in A^d should be removed. From the scenario, we can see that after *t*=2, evacuated node P_0 is removed because it is destroyed by the tsunami. A super-source v_s and a super-sink v_t are added to model the inflow and outflow of evacuees respectively. The algorithm has to find a set of evacuation routes from v_s to v_t that ensure all evacuees reach the safe nodes within the minimum evacuation time. The pseudo code of evacuation route algorithm is as follow:

The following are rules which are applied in this proposed algorithm:

1. All groups of evacuees of an evacuated node *p* should take the same evacuation route ε_c. This is to ensure that there will be no chaos during the evacuation process due to panic evacuees. Panic evacuees could cause conflicted flow of evacuation among groups of evacuees and ineffective evacuation route taken by groups of evacuees due to confusion and other group provocations.
2. An evacuation route ε_p of a group of evacuees that started from an evacuated node *p* is a subset of any evacuation route of a group of evacuees that passes the evacuated node.
3. An evacuation route ε_p should only contain edge *l* at most one.

Input:

- Evacuation graph $G^d = (N^d = P^d \cup T^d, A^d)$, where N^d is a set of evacuated nodes P^d and safe nodes T^d, and a set of directed-edges A^d that connect a node with another node.
- For each node $n \in N^d$, n comprises of (d_i, f_i), where d_i is the deadline for evacuation and f_i is the amount of population that needs to be evacuated.
- For each edge $l \in A^d$, *l* comprises of (s_e, u_e, f_e), where s_e is travel time, u_e is the capacity, and f_e is the time when the edge becomes unavailable.

Output:

- Ω' is an evacuation plan, which contains a set of evacuation routes, with the shortest evacuation time.
- *S* is the evacuation schedule with the maximum number of evacuees.

Algorithms:

Sort node in P^d by its distance to a nearest safe node.

Foreach node $p \in$ *sorted* P^d, generate $\varepsilon<n_0, n_1, \cdots, n_k>$ with the shortest path for each pair of *p* and *t*,

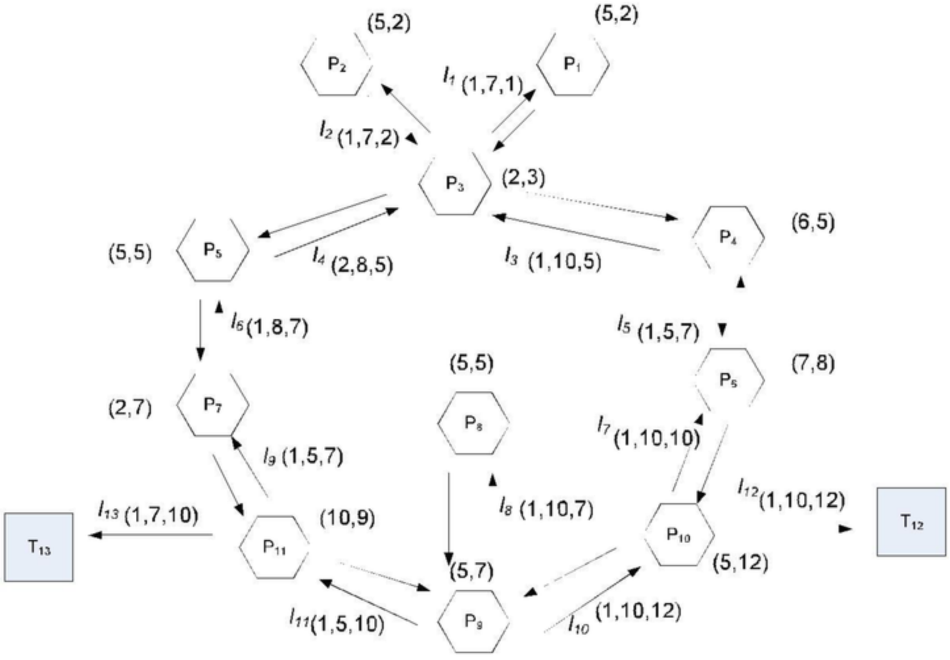


Fig. 3. Experimental Network Topology

```
where $p \in P^d$, $t \in T^d$ generated
While there is an evacuated node $p \in P^d$ where $f_i,>0$ {
    flow = min($f_i,>0$, available_edge_capacity(all edge on route $_c$), available_node_capacity (all node on evacuated node to safe node on route $\varepsilon_c$),
)
for $i$=0 to $k$-1 do {
    $t' = t + \tau_e(e(n_i,n_{i+1}))$;
    available_edge_capacity ($e(n_i,n_{i+1}),t$) - flow;
    available_node_capacity ($n_{i+1}$ , $t$) - flow;
    t = t';
}
$\Omega' = \Omega' \cup G(\Omega', \varepsilon, S)$;
$S = (\Omega' \; G, G^d)$
```

IV. EXPERIMENTAL RESULT

The following are assumptions which are applied in this proposed algorithm:

1. Decision maker instructs where and when each group of evacuees from certain evacuated node must leave and which route should be taken.
2. There should be only one scenario is used at a time.
3. The objective of the decision maker is only to ensure that all evacuees are evacuated to the safe node within a feasible time.

Fig. 3 illustrates the network topology of a coastal area used in the experimentation. The network comprises of 13 nodes which are member of N, that is 11 evacuated nodes P and 2 safe nodes T. The generation process of evacuation plan (Ω') and evacuation schedule (S) are done iteratively until there is no evacuee left in any evacuated node (due to all evacuees have been evacuated or the evacuated node have been drawn by tsunami). When an evacuated node still has evacuees left by the time it was drawn by the tsunami, then a penalty is given.

Fig. 4 visualizes the evacuation plan generated by the algorithm. Table 1 shows the evacuation plan produced by CBMCCRP given the network topology given in Fig. 3.The first two groups of evacuees that started from P_{10} and P_{11} were the first groups of evacuees that reach the safe node t_{12} and t_{13} respectively. Evacuees which started from node P_6 were divided into two smaller groups (groups of 5 and 2 evacuees) which started the evacuation at the same time (t=0) and through the same evacuation route. The first group reached the safe node at t=2, while the second group reached the safe node at t=3. This is because the second group has to wait 1 unit of time at node P10 due to edge capacity of unit l_{12} that connect P_{10} and T_{12}.

The evacuees that started from evacuated node P11 are also divided into two groups of evacuees (groups of 7 and 3 evacuees). The first group departed at t=0, while the second group departed at t=1. Both groups took the

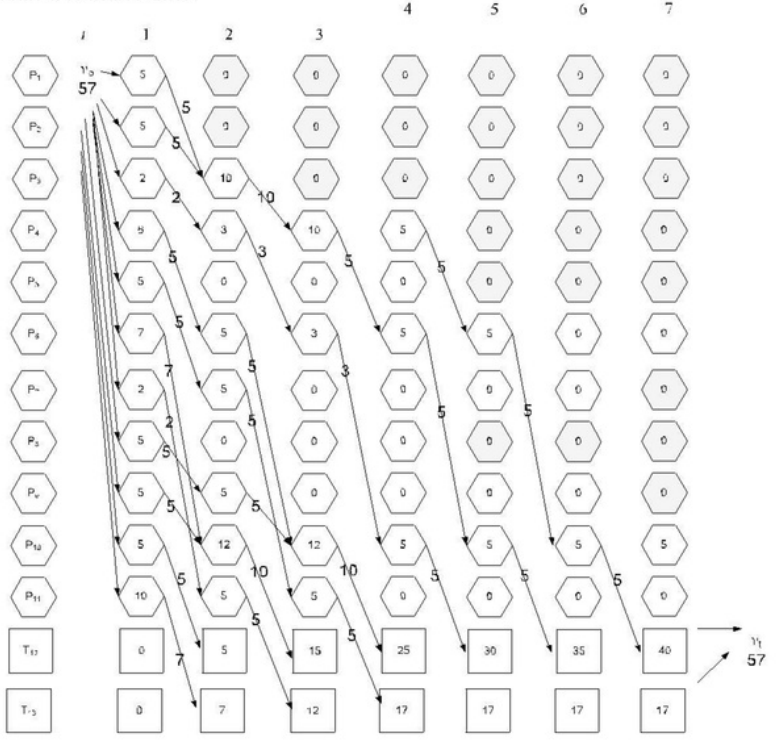


Fig. 4 Generated Evacuation Route of Given Scenario

same evacuation route. The first group that started earlier reach the safe node T_{13} at t=1. The second group that started later joined by another group of evacuees that started from T_7.reach the safe node T_{13} at t=2

The evacuees that started from in evacuated node T_4 were divided into 3 groups (groups of 4, 1, and 1 evacuees). All groups took the same evacuation route, i.e. P_4-I_5-P_6-I_7-P_{10}-I_{12}-T_{12}. The first two groups started at t=0, while the third groups started at t=1. At P_{10}, the first group together with the first group (group of 4) that started from evacuated node P_8 joined the second group of evacuees (group of 2) that started from P_6, which arrived 1 unit of time earlier, and immediately continue travelling to safe node P_{12}. At the same node, after waiting for 1 unit of time due to I_{12} were fully occupied, the second group together with the second group (group of 1) that started from evacuated node P_8 were joined by the third group of evacuees and continue travelling to safe node P_{12}.

Given this result we can see not only the algorithm make sure one route for each evacuated node, but also ensure that an evacuation routeε_a of an evacuated node p_a where p_a is in evacuation route of evacuated node p_b, then $\varepsilon_b \subseteq \varepsilon_a$. Given the problem in Fig. 1, the algorithm produces four paths for generating evacuation routes, that is:

- P_1- I_1-P_3- I_3-P_4- I_5-P_6- I_7-P_{10}- I_{12}-T_{12}
- P_2- I_2-P_3- I_3-P_4- I_5-P_6- I_7-P_{10}- I_{12}-T_{12}
- P_5- I_6-P_7- I_9-P_{11}- I_{13}-T_{13}
- P_8- I_8-P_9- I_{10}-P_{10}- I_{12}-T_{12}

The result shows the different between previous algorithm, i.e. MRCCP, and the proposed algorithm. First, CBMCCRP adopts [4] to generate evacuation path and evacuation schedule separately while ensuring that one evacuated node should only be assigned to one evacuation route.

Second, instead of generating route during the generation of schedule, the algorithm started the process by sort the evacuated node by its distance to the sink-node. The idea is to ensure that the scheduling process start from the closest-to-sink evacuated node. The allocation of evacuees group into the schedule is as follow:

1. Group of evacuees started from the current evacuated node.
2. Group of evacuees which arrived earlier in the current evacuated node.
3. Group of evacuees from the closest evacuation node which arrived in the current evacuated node.

Third, the proposed algorithm splits evacuees based on the capacity of an edge that connect started node to designated node. When a group of evacuees arrived at an evacuated node, where next evacuation edge has less number of available capacities than the number of evacuees to be evacuated, then the group is split into two smaller groups.

Last, MRCCP maintains all routes of each evacuee groups and the capacities of all edges within the routes in every iteration. This is because the algorithm generates new routes based on the capacity of edges at time t. The proposed algorithm does not have to maintain all evacuation routes of all evacuation groups, but the algorithm only has to maintain the main evacuation routes.

Given 11 evacuated nodes, 2 safe nodes, 13 edges, and 57 evacuees, the algorithm required t=7 to evacuate all evacuees to safe nodes. In order to calculate the cost model of the proposed algorithm, we assume that n is the total number of nodes in graph G with n_p is the total number of evacuated nodes and n_g is the total number of evacuee groups generated during evacuation scheduli

TABLE I

EVACUATION SCHEDULE FROM THE SCENARIO

#	Group of Evacuees p_i	Number of Evacuees	Start Time	Route	Exit Time
1	p_{11}	7	0	P11-T13	1
2	p_{10}	5	0	P10-T12	1
3	p_{11}	3	1	P11-T13	2
4	p9	5	0	P9-P10-T12	2
5	p_6	5	0	P6-P10-T12	2
6	p7	2	0	P7-P11-T13	2
7	p6	2	0	P6-P10-*idle*-T12	3
8	p_8	4	0	P8-P9-P10-T12	3
9	p5	5	0	P5-P7-P11-T13	3
10	p4	4	0	P4-P6-P10-T12	3
11	p4	1	0	P4-P6-P10-*idle*-T12	4
12	p8	1	0	P8-P9-P10-*idle*-T12	4
13	p4	1	1	P4-P6-P10-T12	4
14	p3	2	0	P3-P4-P6-P10-T12	4
15	p1	3	0	P1-P3-P4-P6-P10-T12	5
16	p2	3	0	P2-P3-P4-P6-P10-T12	5
17	p1	2	0	P1-P3-P4-*idle*-P6-P10-T12	6
18	p2	2	0	P2-P3-P4-*idle*-P6-P10-T12	6

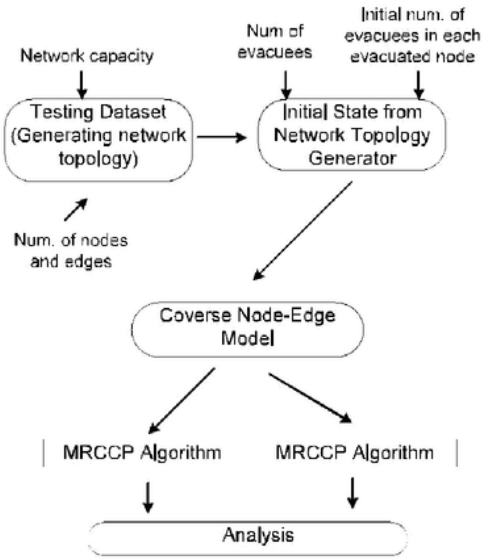


Fig. 5. Scalability Testing Design

The computation model of MRCCP is characterized as iterative problem. In each iteration, an evacuation route of each node is selected and capacities of all edges along the routes are preserved. The number of iterations is determined by the number of groups generated during the evacuation scheduling process. In each iteration, the route with the earliest destination arrival time from started evacuated node is recalculated with the cost of $O(n_p \times n \log n)$. Reservation is made for the node and edge capacities along the chosen route with the cost of $O(n)$. The cost model of MRCCP algorithm is as follows:

$$Cost_{MRCCP} = O\left(\left(n_p \times n \log n + n\right) \times n_g\right) \quad (1)$$

The cost model of CBMCCRP is less complex. The evacuation route is generated once and separated from the calculation of evacuation schedule. Furthermore, the evacuation routes are generated for all evacuation groups that started or passed an evacuated node. Therefore, it decreases the complexity and ensures its scalability for a bigger network topology. The cost model of MRCCP algorithm is as follows.

$$Co\ st_{MR\ CC\ P} = O\left(n_p \times n\ l\ og\ n\right) \quad (2)$$

The number of iteration during the evacuation route generation of CBMCCRP is equal or less than MRCCP algorithm. It is because CBMCCRP ensures that any evacuated node that precedes another evacuated node within the path to sink-node has an evacuation route that includes the evacuation route of preceded evacuation node. Therefore, the path of preceding node should be the addition of evacuated node, connecting edge, and evacuation route of the following node.

Fig 5 shows the testing methodology used for evaluating the scalability of the algorithm. This methodology was adopted from the one used by [11]. The data set used in this testing was adopted from Google Map of Manado area, which lies on the North Coast of North Sulawesi Province, Indonesia. The network consists of 339 edges, 87 evacuated nodes, and 6 safe nodes. The population to be evacuated is 417,787 lives. The distribution of evacuees was based on the number of population in each 87 sub-regions. The test was conducted on Intel i-Core 3 64bit, Portege Z835, 4 GB RAM, Windows 7 Operating System.

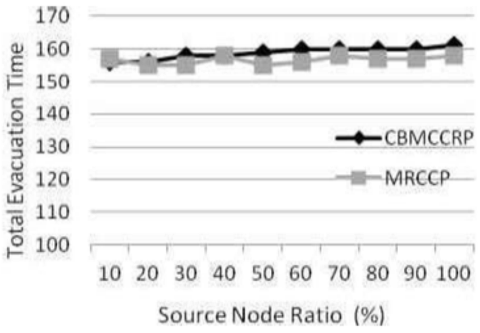


Fig. 6. Evacuation Time vs. Ratio Nodes

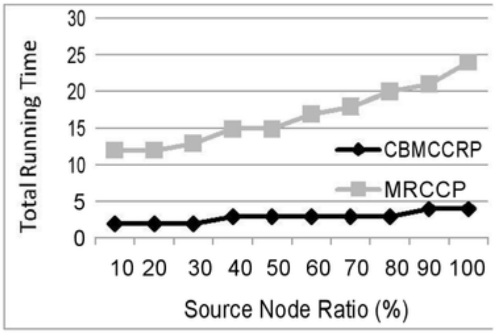


Fig. 7. Running Time vs. Ratio Nodes

Fig 6 and 7 show the result of the scalability testing on MRCCP and CBMCCRP. On one hand, Fig 6 indicates that there is no significant improvement by CBMCCRP with respect to evacuation time. The evacuation time produced by CBMCCRP tends to be longer (1.3 minutes in average) than MRCPP. The evacuation routes generated by CBMCCRP do not change with respect to the changes in the number of evacuees at a certain evacuated node at given time *t*. During the generation of evacuation routes, CBMCCRP adds a penalty to a route that contains evacuated node which becomes unavailable at time *t* while still has evacuees. In MRCCP, the evacuation routes are regenerated in each iteration. When an evacuated node has become unavailable at time *t*, a new column is generated to revise the current route. Two groups of evacuees which start from the same node, at some transit evacuated node may take different paths to the sink-node.

Therefore, the total evacuation time of MRCCP in average is shorter than the one produced by CBMCCRP.

According to Lumbroso et al. [12], the situation during the event of the tsunami disaster is considered times chaotic. Communication tends to be difficult and the command hierarchy structure is fragile and easily fails. This is the result of logistic failure and the nature of the human behavior which is very difficult to predict and control during an emergency situation. This means that it is hard (not to say impossible) to direct various groups of evacuees to take different routes accordingly. Scared and confused evacuees may become misdirected due to various ‘mis’ direction they received from other evacuee groups that they encounter along the route. Stepanov and Smith [13] suggest a clear and easy instruction or information of evacuation route should be given to ensure the optimal result of the evacuation effort.

With respect to computation time, Fig 7 indicates a significant improvement by CBMCCRP compares to MRCCP. It can reduce more than 80% of computation time required by MRCCP. This is because the CBMCCRP produces evacuation route separated from evacuation schedule. The process of generating evacuation routes is relatively linear to the number of evacuated nodes. The generation of evacuation routes is done once, before the generation of evacuation schedule. This approach reduces the complexity significantly.

Separating evacuation route generation and evacuation schedule generation can reduce the time required to produce the evacuation routes for it accelerates the formation of new evacuation route as a remedy to conflicted evacuated route. The separation of evacuation route generation and evacuation schedule generation can reduce the time required to produce the evacuation routes. It accelerates the formation of new evacuation route as a remedy to conflicted evacuated route.

The result of this research can be used by various stakeholder of tsunami disaster in order to produce a tangible and measurable evacuation plan and evacuation schedule of evacuees when an event of big tsunami occurs, especially in a coastal area such as Manado. A good planning can reduce the probability of casualties during the real evacuation event, especially for a coastal area that has dense population such as Manado, Ambon, Jayapura, and Denpasar.

V. CONCLUSION

The proposed algorithm, Conflict-Based Multi-Capacity Constrained Route Planning (CBMCCRP), lowers the computation time required by the previous algorithm, MRCCP, by reducing the complexity of evacuation routes generation. CBMCCRP reduce the complexity by separating the generation of evacuation route from evacuation schedule. . It accelerates the formation of new evacuation route as a remedy to conflicted evacuated route with respect to the dynamic changes of evacuation path availability.

Thus, the total evacuation time is not improved in CBMCCRP. It produce a slightly longer (i.e. 1.3%) evacuation time than MRCCP. Nevertheless, the assumption used in generating the evacuation route is safer, with respect to the chaotic situation during the tsunami event. It can be expected that the evacuation route can work better in real situation.

Further research should be directed to consider other aspects that is relevant to urban areas, such as dynamic population distribution with respect to time (working hours, rushing hours, rest hours, etc.) and day (working days and weekends), building structures, the role of social network applications and devices, etc. Complex variables may improve the model precision to the near-real life situation.

IAENG_W

ORIGINALITY REPORT

10%	4%	6%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	connection.ebscohost.com Internet Source	4%
2	Victor Pillac, Pascal Van Hentenryck, Caroline Even. "A conflict-based path-generation heuristic for evacuation planning", Transportation Research Part B: Methodological, 2016 Publication	3%
3	"Intelligence and Security Informatics", Springer Nature America, Inc, 2003 Publication	1%
4	Qingsong Lu. "Evacuation Planning: A Capacity Constrained Routing Approach", Lecture Notes in Computer Science, 2003 Publication	1%
5	Lecture Notes in Computer Science, 2008. Publication	<1%
6	Balakrishnan, N.. "Optimal step-stress testing for progressively Type-I censored data from exponential distribution", Journal of Statistical	<1%

Planning and Inference, 20090501

Publication

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

IAENG_W

GRADEMARK REPORT

FINAL GRADE

/0

GENERAL COMMENTS

Instructor

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7