Resource Delivery Path Dependent Deployment Scheduling for Contingency Cellular Network

Yao-Nan Lien, Tsai-I Kao [1]
Department of Computer Science[1]
National Chengchi University
Taipei, Taiwan, R.O.C.
{lien,g10031}@cs.nccu.edu.tw

Jyh-Shyan Huang [1,2]
Business Customer Solutions Lab.[2]
Chunghwa Telecom Laboratories
Taipei, Taiwan, R.O.C.
frank210 @cht.com.tw

Abstract*—When a catastrophic natural disaster occurs, the efficiency of disaster response is crucial to life saving. However, mobile communication systems usually crashed due to various causes, making the coordination among a large number of disorganized disaster response workers extremely difficult. We propose a Contingency Cellular Network (CCN) by connecting disconnected base stations together with wireless links and portable power generators. Since the transportation capacity may be very limited, scheduling of CCN deployment order according to the demand of disaster response becomes an important issue. Two models aiming to maximize disaster response efficiency are proposed in this research. Emergency levels of disaster areas, time decreasing profit functions, traveling time of delivery path, and antecessor precedence constraint are taken into consideration in the CCNDS-AC model. The second model, CCNDS-UC is the same but with antecessor precedence constraint relaxed. Both problems are proven NP complete. Two heuristic but efficient algorithms are designed to solve the problems. Both algorithms can outperform our previous algorithm by a significant margin in our simulation experiments. This shows that the traveling time is a significant factor in designing an effective CCN deployment scheduling problem.

***Keywords*—Disaster Management, Emergency Communication, Mobile Communication, Ad Hoc Network, Deployment Scheduling**

I. INTRODUCTION

A. *The Impact of Large-Scale Natural Disasters*

Frequently occurring large-scale natural disasters have been reported to cause great damage in recent years, claiming many people's lives, rendering millions people homeless, and causing huge financial loss. The Haiti Earthquake that occurred in 2010 alone claimed 230,000 lives.

Take the Japan Northeastern Earthquake that happened on March 11, 2011 as an example, a 9.0 magnitude earthquake, followed by a 23-meter high tsunami and a nuclear melt-down disaster together created a complex major natural disaster stunning the whole world. We summarize the common problems of large-scale natural disasters, such as 921 Chi-Chi Earthquake (Taiwan) [3], 88 Flood (Taiwan), SiChuan (China) [9] and Haiti Earthquakes as follows:

- large scale of buildings collapse
- paralysis of transportation system
- paralysis of communications network
- lack of professional disaster response workers
- dysfunctional administrative command system

The impact to the disaster response caused by above factors are as follows:

- difficult to transport resources to the disaster areas
- inefficient resource allocation and reallocation resulting in misplacement of resources
- inefficient coordination among disaster response workers lowering disaster response efficiency

Although plenty of resources may be available from all over the world to assist a large scale natural disaster, due to the above mentioned inefficiency, a disaster may still claim many lives that could have been saved if available resources could be used in a more efficient way. A well operated mobile communication system is certainly a key factor to improve the efficiency of a disaster response operation.

B. *Communication Systems Crash*

It has been known for a long time that a communication system is crucial to disaster response. However, many of seemingly stable public communication networks did not survived in previous disasters. Surprisingly, we found that during 88 Flood and 921 Chi-Chi Earthquake in Taiwan and Hurricane Sandy in the East coast of United State, the cell phone networks were vulnerable due to the following reasons:

Service disruption of base stations: Common reasons are (1) power outage (the backup batteries usually can only last several hours); (2) broken backhaul; and (3) physical destruction by earthquakes/flood/hurricane.

Critical hardware equipments were knocked down: Due to (1) external power outage; (2) fuel for power generator exhausted; (3) cooling system broken; and (4) switch overheated.

Sponsored by National Science Council NSC 101-2420-H-004-006-MY3

The cables of power lines or backhaul links are usually laid along roads and bridges for the convenience of deployment and maintenance. The destruction of roads and bridges, which was a common phenomenon in a disaster, leads to power outage and network disconnection.

Although power lines and communication backhauls usually have redundancy for higher availability, they may not necessary improve the survivability significantly in a large scale disaster. For instance, a huge flood over a river may destroy many bridges over the river at the same time breaking all redundant cables completely. The backhaul of base stations, the basic structure of mobile communication systems, must be connected to the controllers or switches. Even if a base station remains intact in a disaster, as long as its backhaul is disconnected, it can no longer keep in operation.

Take 88 Flood for example, the structure of many base stations remained intact and free of flood because they were often located on a higher place. However, when the power lines and backhauls that were laid along the roads and bridges were destroyed, mobile communications system was therefore paralyzed. Power lines and backhauls become an Achilles' heel of many existing mobile communications networks.

Encumbered by the above mentioned and many other problems, it is prohibitively expensive to build stronger mobile communication networks. Although National Communications Commission of Taiwan built a number of strengthened base stations with satellite communication for backhauls. Unfortunately, the number of such base stations is very limited due to fund constraint.

C. Requirements of Contigency Comunication Network

Based on the analysis shown above, we summarized a set of system requirements in [4], called 7-ability. Among them, the two most ignored by existing contingency communication systems are terminal popularity and practicality. No existing system can afford a large number of mobile terminals for all potential users. Furthermore, due to the lack of commercial incentive, only very limited fund can be devoted to the development of such a system such that it is not practical to design a complete new system from scratch.

D. Contingency Cellular Network

We designed an emergency communication network for disaster response, called *Contingency Cellular Network* (CCN) [4,5,6] by connecting disconnected base stations in the disaster areas with wireless links and portable power generators. Such a system will be able to support many voluntary workers and victims in the early hours of catastrophic natural disasters, and thus may save many lives.

Design philosophy of CCN is to reuse existing disconnected base stations in the disaster area. The reasons are as follows: (a) wide coverage of mobile communication network; (b) widespread use of cell phones; (c) only a low cost add-on module is needed to repair a disconnected base station; (d) low-barrier of usage, which are mostly needed in disaster response communication. One crucial non-technical reason is that cell phone might be the first thing carried by most victims and people who escape from their homes when a disaster strikes. Therefore, reconnecting disconnected base stations in the disaster area to provide a low-cost large-scale emergency communication service is a good option.

Contingency Recover Package (*CRP*) consists of a power module, a number of Inter-Cell Communication Module (*ICC Module*), and an add-on processing module, which is referred to as Emulated Controller Module (*EC Module*). CRP can be stored in national disaster response centers or cellular operators and delivered to the selected base stations via airdrops or helicopters. The EC-Module is connected to a base station in the first step. Then, ICC Modules are used to connect the base station to its neighbors in the second step via long range wireless links. At lease a pair of ICC Modules is needed for each base station. A multi-hop wireless network overlapped on top of the selected base stations is finally formed. The overlapped network provides the connectivity between base stations and core network. Anyone who has a cell phone can access service from these base stations. If there is no way for a disconnected base station to connect to the core network, some CRP may include a Satellite Communication Module (SC Module) to be used to establish a connection to the core network.

Challenges, related researches, design philosophy and system architecture of CCN were discussed in paper [1,4,5,6,8].

II. RELATED RESEARCHES

In the planning phase of a CCN construction, there are several issues needed to be addressed. The first is to select the base stations out of all disrupted base stations based on the number of available CRPs and to determine the topology. The second is to determine the deployment schedule to deliver the CRPs to the selected base stations for restoration.

Network Topology Planning Algorithm [6] was proposed to select base stations and to compute the CCN *forwarding tree* (FT). A survival base station is chosen as the root. And thus, other stations can connect to the core network by multi-hop connections through the root. The forwarding tree is re-planned immediately, if any link in the forwarding tree is broken.

Since the transportation capacity may be very limited, it may need several rounds of CRP delivery. Unfortunately, the benefit of saving a station is gradually attenuated with time such that the deployment sequence will largely determine the disaster response efficiency. A good deployment sequence can save more lives than what a bad one can do. The determination of deployment sequence is similar to the conventional scheduling problems but with an extra constraint that ancestor nodes have to be rescued before their descendent nodes, because an ancestor node has to forward the traffic for its descending nodes.

A CCN deployment scheduling (CCN-DS) formulation as well as a heuristic deployment scheduling to approximately maximize the efficiency of disaster response were proposed in [7]. Unfortunately, that paper didn't take the traveling time of each selected path into account. This paper is trying to improve the CCN-DS model by taking traveling time into account.

The rest of this paper is organized as follows. Section III presents the new CCN deployment scheduling problem. Experiments are provided in Sections IV. Discussion of concluding remark and future work is in Section V.

III. RESOURCE DELIVERY PATH DEPENDENT DEPLOYMENT SCHEDULING

A. *Definition of Profit*

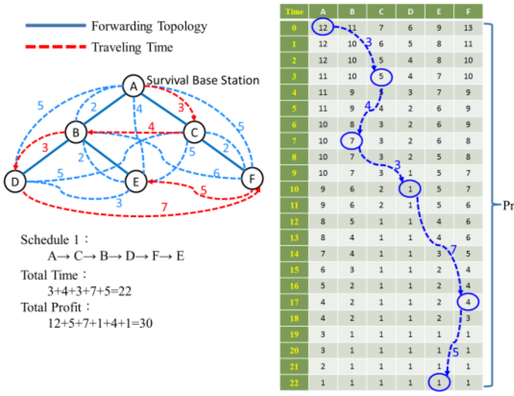
The emergency level of the area covered by a base station is represented by a time-variant profit parameter, which has to be defined by the disaster response authority because they have not only the necessary knowledge for disaster response, but also the official situation statistics. A typical example of profit definition is the estimated time dependent survival rate.

B. *Problem Models*

The Resource Delivery Path Dependent CCN Deployment Scheduling Problem is formulated into two optimization models: the first one, CCNDS-AC, is with Antecessor Precedence Constraint, and the second one, CCNDS-UC, is without the constraint. We assume one or more forwarding tree is calculated in advance. (Note that a multiple forwarding tree problem can be easily converted into a single forwarding tree problem by adding an extra virtual root connecting all forwarding tree together.)

The non-preemptive CCNDS-AC is as follows. A set of nodes organized in a tree structure has to be fixed by work teams; a preceding node must be rescued before its descendants; the profit of fixing a node is a function of time; the traveling time from each node to every other node is also given; the CCNDS-AC problem is to find a deployment sequence such that the total profit is maximized. An example is shown in Fig. 1.

The graph on the left of Fig. 1 is the forwarding tree. The blue edges are the wireless links and the red paths are traveling paths labeled with traveling time. The table on the right shows the deployment sequence labeled with traveling time. The value in each table cell is a time-variant profit.



Time	A	B	C	D	E	F
0	12	11	7	6	9	13
1	12	10	6	5	8	11
2	12	10	5	4	8	10
3	11	10	5	4	7	10
4	11	9	5	3	7	9
5	11	9	4	2	6	9
6	10	8	3	2	6	9
7	10	7	3	2	6	8
8	10	7	3	2	5	8
9	10	7	3	1	5	7
10	9	6	2	1	5	7
11	9	6	2	1	5	6
12	8	5	1	1	4	6
13	8	4	1	1	4	6
14	7	4	1	1	3	5
15	6	3	1	1	2	4
16	5	2	1	1	2	4
17	4	2	1	1	2	4
18	4	2	1	1	2	3
19	3	1	1	1	1	1
20	3	1	1	1	1	1
21	2	1	1	1	1	1
22	1	1	1	1	1	1

Fig. 1 Example of deployment schedule

Mathematical model of CCNDS-AC is as follows. Given a forwarding tree, $T(V,E)$, where

- $V = \{v_i \mid i=0,1,2,\ldots,n\}$, is the set of base stations and v_0 is the root which has an external link;
- $E = \{e_{i,j} \mid e_{i,j}, v_i, v_j \in \mathrm{V}\}$is the set of link;
- $R = \{r_i \mid i=1,2,\ldots,n$ and $r_i > 0\}$ is the set of construction time of the isolated station v_i.
- $\pi = \{\pi_a \mid \pi_a = (\pi_a(1),\ldots,\pi_a(n))$, a is a positive interger$\}$ is the set of CCN deployment schedules. $\pi_a(i)$ is the position of v_i in schedule π_a.
- $D = \left\{d_{ij} \middle| v_i, v_j \in V\right\}$ is the set of traveling time;
- $P = \{P_i(t) \mid v_i \in V\}$ is the set of profit, $P_i(t)$ is the profit of v_i when it is constructed at time t. With respect to a schedule π_a , the construction time of v_i is $\sum r_{\pi a(k)}$ where k is a node precedes node v_i in schedule π_a.

The objective of CCN deployment scheduling problem is to find a deployment schedule π_a from π, to

maximize

$$\sum P_i(t) = \sum P_i(c(\pi_a,i)),\ \ i=1,2,\ldots,n$$

subject to

$\pi_a(i)$ precedes $\pi_a(j)$, where v_j is the descendant of v_i for all $v_i, v_j \in V$.

For the sake of discussion, an *isolated node* is defined as a node whose parent node hasn't been rescued (visited). The antecessor precedence constraint in CCNDS-AC forces the transportation vehicle to ignore any isolated node even if the vehicle passes such a node. It might be beneficial to visit such an isolated node without any extra traveling cost. Even though a rescued isolated node is not able to provide any service immediately after it is rescued, it can save a rescue trip after its parent node is rescued. Taking the example shown in Fig. 2, the red path in the left graph is a solution of CCNDS-AC and in the right graph is a solution ignoring antecessor precedence constraint. Both paths started from the headquarter that is located near node E. As we can see that the traveling time of the path visiting node A, B, and E on the right graph is smaller than its counterpart on the left graph. Take this consideration, we propose another model, Unconstrained CCN Deployment Scheduling Problem (CCNDS-UC) which is the same as CCNDS-AC but without antecessor precedence constraint.

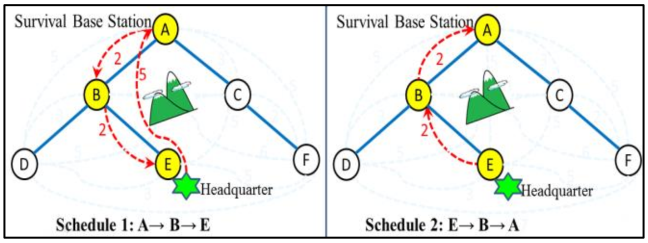


Fig. 2 A path ignoring antecessor precedence constraint.

C. Complexity Analysis

Similar to the conventional machine scheduling problem, CCNDS-AC can be easily proven to be a NP-hard problem.

CCNDS-AC is in NP：

We first show that CCNDS-AC ϵ NP. Assuming that we are given a forwarding tree $T(V,E)$, as well as a schedule, we can use a double loop to verify that a parent node must be visited before its child nodes in T. The verification algorithm can affirm the schedule is a valid CCNDS-AC schedule within O(n^2) time.

CCNDS-AC is NP-Hard：

We now prove that CCNDS-AC problem can be reduced to the single machine scheduling problem (SMS) straightforwardly.

SMS is defined as follows: Given a set J of n independent jobs that has to be scheduled on a single machine. Each job $j_i \in J$ contains uninterrupted processing time $u_i \in U$ and weight $w_i \in W$, where u_i and w_i are positive integers. The single machine can handle only one job at a time.
SMS is to find a schedule π such that $\sum_{i=1}^{n}(w_i * C_i(\pi))$ is minimized, where π is a permutation of all jobs ($k = 1, 2, 3, \dots, n!$), $C_i(\pi)$ is the time at which job j_i completes in the given schedule π.

Given an instance A:[J,W,U] in SMS, we can find an instance B:[V,E,R,D,P] with a single-level Forwarding Tree in CCNDS-AC such that an optimal solution π_b for B is also an optimal solution for A. Let V=J, D={0|all paths}, R=U, $P=\{-w_i t, \mid w_i \in W\}$, $E=\{e_{\mathrm{root,i}}|v_{\mathrm{root}}, v_{\mathrm{i}} \in V\}$. The verification can be performed in polynomial time. Let total weighted completion time of SMS for π is TWC(π) = $\sum_{i=1}^{n}(w_i * C_i(\pi))$, total weighted profit of CCNDS-AC for π is TWP(π) = $-$TWC(π). We prove the following 3 Lemmas first：

Lemma 1: Any valid schedule π_b for B(CCNDS-AC) is a valid solution for A(SMS).

Proof: Any permutation of J is a valid schedule for A, and π_b is a permutation of V, which is equal to J. Therefore π_b is a valid solution for A. Q.E.D.

Lemma 2: Any valid schedule π_a for A is also a valid schedule for B.

Proof: Any given schedule π_a for A is a permutation of J which is equal to V. Therefore, π_a is a permutation of V. Since each node in B can directly connect to the root of B such that the ancestor precedence constraint is always non-existing. Therefore, π_a, a permutation of V, is a valid schedule of B. Q.E.D.

Lemma 3: If TWC(π_a) < TWC(π_b), then TWP(π_a) > TWP(π_b).

Proof: If $\sum_{i \epsilon J}(w_i * C_i(\pi_a))$ < $\sum_{i \epsilon J}(w_i * C_i(\pi_b))$, by Equal Division Theorem, we can get $\sum_{i \epsilon N}(w_i / C_i(\pi_a))$ > $\sum_{i \epsilon N}(w_i / C_i(\pi_b))$. Q.E.D.

Next, we prove by contradiction that an optimal solution π_b to B must be an optimal solution to A. By Lemma 1, we know π_{b} is also a valid schedule for A, whose total weight completion time is TWC(π_b). Assume π_b is not an optimal schedule for A, there must be another schedule π_a, whose total weighted completion time TWC(π_a) is smaller than TWC(π_b). By Lemma 2, π_a is also a valid schedule for B, whose total weighted profit is TWP(π_a). By Lemma 3, we can obtain TWP(π_a) is bigger than TWP(π_b). This contradicts to the fact that π_b is an optimal solution for B. Therefore, π_b must be an optimal solution for A. Q.E.D.

Similarly, CCNDS-UC can also be proven to be NP-hard in a similar way.

D. Heuristic DS-ACG Algorithm

Sine CCNDS-AC is an NP-hard problem, we designed a heuristic approximated algorithm, DS-ACG, to solve it. The algorithm is basically a greedy algorithm. The antecessor precedence constraint effectively reduces the number of choices in each iteration of the algorithm. Therefore, a greedy algorithm might perfectly fit the problem itself and obtain a very good performance.

In the algorithm, a forwarding tree $T(V,E)$, a set of nodes π_a (SL), and all input parameters are given initially. The candidate list (CL) is initialized to the children of the root. In each iteration, the DS-ACG algorithm chooses the node from CL which has the maximum profit to attach to the tail of π_a. The children of the newly selected node are included to CL. The procedure is repeated iteratively until CL is empty. The time complexity of DS-ACG is $O(n^2)$. Although the algorithm is rather simple, it outperforms easily our previous algorithm DS-G because DS-ACG takes into account the traveling time. The pseudo code is shown as follows.

DS-ACG Pseudo code:

```
Input CCN Topology, Traveling Time, Profit
SL=null        /*the Scheduling List*/
```

```
CL={}            /*the Set of Candidate nodes*/
CL.add(Survival BS);
while (size of CL !=0) do
    for each node in CL do
        calculate the node's profit and time
        if (profit/time) >(maxProfit/maxTime)
            maxProfit = profit
            maxTime = time
            maxPtofitNode =node
        end if
    end for
    SL=SL+ maxPtofitNode                /*update Scheduling List*/
    totalProfit=totalProfit+maxProfit   /*update total Profit*/
    totalTime=totalTime+maxTime         /*update total Time*/
    CL.remove(maxPtofitNode)
    for each node in maxPtofitNode's childList do
        CL.add(node)
    end for
end while
```

E. Heuristic DS-UCB Algorithm

Because of extremely stringent time constraint in disaster response, most heuristic solutions we developed so far are polynomial time greedy algorithms, which iterately select the best choice, which has the largest estimated profit, based on the current state without backtracking. Heuristic algorithms often offer near optimal solutions because the number of choices is usual very limited due to ancester precedent constraint. Unfortuantely, because ancester precedent constraint is relaxed in CCNDS-UC, not only the number of choices in each iteration is much more than that of CCNDS-AC, but also the estimated profits for isolated nodes are unknown unless a looking-ahead and backtracking computation steps is included in the algorithm. A greedy algorithm will perform poorly under such a condition. Therefore, we propose DS-UCB algorithm that is basically a modification of DS-ACG with a limited looking-ahead and backbacking mechanism. DS-UCB is still an approaximated algorithm without any gurantee of optimality. Similar to DS-AC Each iteration of DS-UCB algorithm consists of two major steps: the first step is to find a temprary path, which is actually a temparary subschedule, from the current selected node upward to a node whose profit is computable; then the second step is to compute the best path from the head to the tail of the path among all possible paths within three hops of the temprarly path. The selected best path is the new subschedule to be added to the tail of the current schedule. Set the node nearest to the current node to be the new current node. Intial current node is the the node nearest to the headquater. The iteration is repleated until all nodes are included in the schedule. The time complexity is O(n^3). However, it is still a polynomial complexity and will be acceptble if the number of nodes is not too high, say, under 1000. Note that the topology in the real world will not be fully connected so that the complexity will never reach 1000^3). The pseudo code is shown as follows.

DS-UCB Pseudo code:

```
Input CCN Topology, Traveling Time, Profit
SL=null          /*the Scheduling List*/
CL={}            /*the Set of Candidate nodes*/
startNode=the closest node of Headquarter
while(SL.length!=node Number of CCN topology) do
    SL.add(startNode)
    if(startNode don't have profit) do
        /*initial path from start node to SBS*/
        while(startNode don't have profit) do
            temp_Sch.add(parentNode)
            startNode=parentNode
        end while
        backtrack_Sch=Backtrack(temp_Sch);
        max_Sch=temp_Sch
        if profit(temp_Sch) <profit(backtrack_Sch) do
            max_Sch=backtrack_Sch
        end if
        totalProfit=totalProfit+ profit(max_Sch)   /*update total Profit*/
        totalTime=totalTime+ time(max_Sch)         /*update total Time*/
        SL.add(max_Sch)
        startNode =last node of SL
    else
        totalProfit=totalProfit+ profit(startNode)   /*update total Profit*/
        totalTime=totalTime+ time(startNode)         /*update total Time*/
    end if
    startNode=the closest node of startNode
end while

Backtrack(temp_Sch) do
    Check temp_Sch neighbor and generate finite schedules
    Compare schedules' total profit
    Choose the schedule with largest total profit as the backtrack_Sch
    return backtrack_Sch
end Backtrack
```

IV. PERFORMANCE EVALUATION

The two proposed CCN deployment scheduling algorithms, DS-ACG and DS-UCB were evaluated against our previous scheduling algorithm DS-G and optimal solution by simulation.

A. Profit Function

The profit function of a base station is assumed a two-segment piecewise linear time-variant function. The initial profit is x at disaster time and decreases with time. The slope from disaster time to P is $s1$ and becomes $s2$ after time P. P is called the *turning point* of profit. Their values were populated randomly as shown in Table I.

B. Expermental Enviroment

Test instances were generated by uniform random functions. The ranges of values used in Experiment I are shown in TABLE I. In Experiment II, the same set of parameters was used to generate test cases except that the size of graph is 50. In this size, there is no way to obtain optimal solutions for such a NP-hard problem. Therefore, we took the best solution among 10 million solutions as the pseudo optimal solution to evaluate the performance of our algorithms. The evaluation metrics are total profit, total traveling time, original profit deviation and normalized profit deviation as shown in (1) and (2).

TABLE I. Parameters of test instances in Experiment I

Parameters	**Range of values**
Initial Profit	x ~ Uniform(30, 100)
Traveling Time	d ~ Uniform(0.5, 10) hr
Turning point	P ~ Uniform (0, 168) hr
Slope1	s1 **~** Uniform (-1, 0)
Slope2	s2 **(|s2|>=|s1|) ~** Uniform (-1, 0)
Turning Point	r_i ~ Uniform (50, 80) hr
Forwarding Tree Size	8-12

$$\text{Original Deviation} = 1 - \frac{\text{Algorithm solution}}{\text{Optimal solution}} \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

$$\text{Normalize Deviation} = \frac{\text{Optimal solution} - \text{Algorithm solution}}{\text{Optimal solution} - \text{Worst solution}} . (2)$$

C. Experimental Results

The results of Experiment I and II are shown in Fig. 3 (only the case of 12 nodes is shown) and Fig. 4. As we can see from these figures that both DS-ACG and DS-UCB outperform DS-G by a very large margin in both small and large cases. These results show that the traveling time is a significant factor in deployment scheduling and cannot be ignored. Furthermore, DS-UCB performs the best among all three heuristic algorithms. This shows that relaxing the antecessor precedence constraint is beneficial. Finally, the normalized deviations from optimal solutions in terms of total profile and total time are all nominal, under 5%, which is not shown here for space saving. These results show that our heuristic algorithm is effective although they are simple.

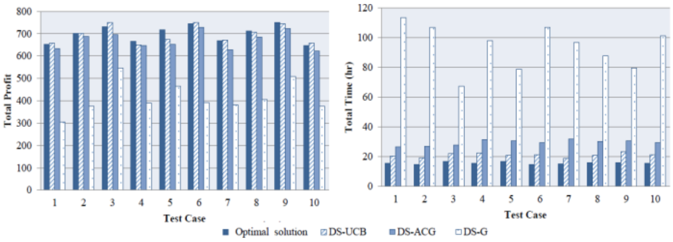


Fig. 3 Total profit and time in Experiment I

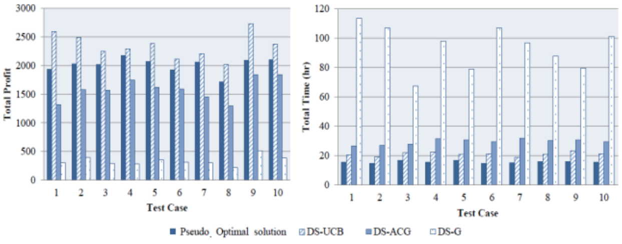


Fig. 4 Total profit and time in Experiment II

V. CONCLUGIND REMARK AND FUTIRE WORKS

We propose a Contingency Cellular Network (CCN) by connecting disconnected base stations together with wireless links and portable power generators. CCN can support existing cell phone users in a large scale natural disaster with very low cost and rapid deployment time. Since the transportation capacity may be very limited in a disaster area, scheduling of CCN deployment order according to the demand of disaster response operation becomes an important issue. We proposed two optimization models aiming to maximize the disaster response efficiency. Both models take traveling time into account, but one does have antecessor precedence constraint and the other doesn't.

Both problems are proven NP complete problems so that we proposed two heuristic algorithms, DS-ACG and DS-UCB, to solve the problems in limited time. Both algorithms were evaluated using simulation. From our experiments, we can see that both algorithms outperform our previous algorithm by a very large margin. This proves that traveling time has a significant impact to the effectiveness CCN of deployment scheduling problem.

In the future, we will develop and implement the components of network part of EC Module and use wireless access points as ICC Module to build the simulation environment of CCN network to verify the CCN design concept.

REFERENCES

[1] Y. Bai, W. Du, Z. Ma, C. Shen, Y. Zhou, and B. Chen, "Emergency communication system by heterogeneous wireless networking," in Proc. of IEEE Wireless Communications, Networking and Information Security (WCNIS), June 2010.

[2] Ling-Yeu Chung, Ta-Yuan Chou and Chung-Chieh Lee, “Multiobjective Dynamic Length Genetic Algorithm to Solve the Emergency Logistic Problem”, International Conference on Advanced Mechatronics, July 2012

[3] Weimin Dong, et al., Chi-Chi, Taiwan Earthquake Event Report, Risk Management Solutions, Inc., https://www.rms.com/Publications/Taiwan_Event.pdf, retrieved Mar. 2010.

[4] Jyh-Shyan Huang and Y.N. Lien, "Challenges of Emergency Communication Network for Disaster Response", in Proc of IEEE International Conference on Communication Systems, Nov. 2012, Singapore.

[5] Jyh-Shyan Huang, Y.N. Lien, C.L. Hu, “Desing of Contingency Cellular Network,” Proc. of APNOMS2012, Sep. 2012, Korea.

[6] Jyh-Shyan Huang, Y.N. Lien and Y.C. Huang, "Network Topology Planning for Contingency Cellular Network", *Proceedings of 17th Mobile Computong Workshop*, Aug. 2012, Taiwan.

[7] Jyh-Shyan Huang, Yan-Song Wang, and Yao-Nan Lien, "Deployment Scheduling of Contingency Cellular Network for Disaster Relief Operations", to appear in the 15th Asis-Pacific Network Operations and Management Symposium, Sep. 2013, Hiroshima, Japan.

[8] Zhenhong Shao; Yongxiang Liu; Yi Wu; Lianfeng Shen, “A Rapid and Reliable Disaster Emergency Mobile Communication System via Aerial Ad Hoc BS Networks,” in Proc of IEEE Wireless Communications, Networking and Mobile Computing (WiCOM), Sept., 2011.

[9] Yang Ran, "Considerations and Suggestions on Improvement of Communication Network Disaster Countermeasures after the Wenchuan Earthquake", IEEE Communications Magazine Jan. 2011.