

Numerical optimization applied to crisis management : A hybrid approach for solving the integrated problem of scheduling and resource allocation.

Mohamad Khorbatly

▶ **To cite this version:**

Mohamad Khorbatly. Numerical optimization applied to crisis management : A hybrid approach for solving the integrated problem of scheduling and resource allocation.. General Mathematics [math.GM]. Normandie Université, 2018. English. NNT : 2018NORMLH18 . tel-01946726

HAL Id: tel-01946726

https://theses.hal.science/tel-01946726

Submitted on 6 Dec 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THESE

Pour obtenir le diplôme de doctorat

Spécialité : **Mathématiques Appliquées et Informatique**

Préparée au sein de « Université Le Havre Normandie - LMAH »

Optimisation numérique appliquée à la gestion de crise : Approche basée sur un algorithme hybride pour la résolution du problème intégré d'ordonnancement et d'allocation des ressources.

Présentée et soutenue par
Mohamad KHORBATLY

Thèse soutenue publiquement le 24 octobre 2018 devant le jury composé de		
M. Henri BASSON	Professeur, Université du Littoral - Côte d'Opale (ULCO)	Rapporteur
M. Abderrafiaa KOUKAM	Professeur, Université de Technologie de Belfort-Montbéliard	Rapporteur
M. Cyril FONLUPT	Professeur, Université du Littoral - Côte d'Opale (ULCO)	Examinateur
Mme Marianne DE BOYSSON	Maître de Conférences, Université Le Havre Normandie	Examinatrice
M. Abdallah EL CHAKIK	Maître de Conférences, Beirut Arab University, Lebanon	Examinateur
M. Hassan ALABBOUD	Maître de Conférences, Université Libanaise, Tripoli – Liban	Co-encadreur de thèse
M. Adnan YASSINE	Professeur, Université Le Havre Normandie	Directeur de thèse

Thèse dirigée par Adnan YASSINE, **Laboratoire de Mathématiques Appliquées du Havre (LMAH), Université Le Havre Normandie – France.**

Etablissement **Ecole Doctorale** **Laboratoire**

Dédicace

A mon père, Fawzi, pour sa patience, ses précieux conseils, les valeurs et principes qu'il m'a éduqués, et tous ses sacrifices ;

A ma douce maman, Hadla, pour tout son amour, sa tendresse et son affection sans limites ;

A mon épouse, Zeinab, pour son amour, sa présence et son soutien sans faille, et à qui je dois tout ;

A mes très chères filles, Jana et Nour pour tous les bons moments passés ensemble ;

A ma très chère sœur Safa, pour tout ce qu'elle a fait pour moi ;

A mes frères Mahmoud et Ahmad, pour leur encouragement ;

A tous ceux et celles qui me sont chers, qui ont participé de près ou de loin à l'aboutissement de ce travail.

Remerciements

A Dieu, qu'il soit béni et exalté, nous adressons les plus chaleureux remerciements. Jamais les mots ne pourraient révéler ce que nous ressentions envers notre créateur. Celui qui a voulu, par sa générosité, couronner un long chemin d'endurance par la bonne réussite.

Je tiens à remercier le directeur de ma thèse, le Professeur Adnan YASSINE, pour son aide inestimable, sa disponibilité, ses encouragements et ses précieux conseils qui ont permis de mener à bien ce travail.

J'adresse aussi mes reconnaissances à mon encadreur Dr. Hassan ALABBOUD qui, depuis quelques années ses conseils et ses connaissances m'ont bien aidé à avancer mes travaux de recherche.

Je souhaite également exprimer ma gratitude à Monsieur Abderrafiaa KOUKAM, Professeur à l'Université de Technologie de Belfort-Montbéliard, et Monsieur Henri BASSON, Professeur à l'Université du Littoral - Côte d'Opale, de m'avoir fait l'honneur d'accepter de rapporter le travail de ma thèse.

Je tiens aussi à remercier Mme. Marianne DE BOYSSON - Maître de Conférences – Université Le Havre Normandie, Monsieur Cyril FONLUPT, Professeur à l'Université du Littoral - Côte d'Opale et Monsieur Abdallah EL CHAKIK - Maître de Conférences à l'Université Arabe de Beyrouth, pour avoir accepté juger mon travail.

Mes remerciements vont tout particulièrement à Dr. Hamdi DKHIL et au Dr. Mohamad AL FALOU pour leurs encouragements continuels et ses conseils de didacticien avisé.

Je voudrais aussi exprimer ma gratitude envers tous ceux qui m'ont accordé leur soutien, tant par leur gentillesse que par leur dévouement.

Je ne peux nommer ici toutes les personnes qui de près ou de loin m'ont aidé et encouragé, mais je les en remercie vivement.

Nous voudrions leur adresser nos remerciements pour nous avoir épaulés tout le long des années d'études et pour ne jamais manquer de nous encourager et de nous soutenir.

Table de matières

Contents

Résumé 1

Introduction Générale: 3

Plan synthétique de la thèse. 4

FIRST PART: LITTERATURE REVUE AND THESIS POSITIONNING 11

Chapter I – Preparation and general definitions 15

1.1 Introduction 15

1.2 Logistics and supply chain 15

1.2.1 Transport logistics 16

1.2.2 Crisis management supply chain 17

1.2.3 Supply chain 17

1.2.4 Supply chain structure 19

1.2.5 Control 19

1.2.5.1 Centralized control, conventional centralized systems: 19

1.2.5.2 Evaluation extent in a state of crisis 20

1.3 Optimization problem 20

1.4 Mathematical modeling encountered problems 21

1.4.1 Linear model 21

1.4.2 Binary linear model 22

1.4.3 General principle of cutting methods 22

1.5 Exact construction methods 22

1.5.1 Branch & Bound (B&B) 22

1.5.2 Branch and Cut 23

1.5.3 Simplex method 24

1.5.4 Dynamic programming 24

1.6 Heuristics and meta-heuristics search 25

1.6.1 Simulated Annealing 25

1.6.2 Tabu Search 26

1.6.3 Ant colony optimization 28

1.6.4 Scalable Algorithms 30

1.6.4.1 A Algorithm* 30

1.6.4.2 Genetic Algorithms 30
1.6.4.3 Greedy algorithms 34
1.6.4.4 Greedy Randomized Adaptive Search Procedure (GRASP) 36
1.6.4.5 Hybrid algorithms 36
1.7 Conclusion 38
Chapter II – Thesis Positioning 41
2.1 Introduction 41
2.2 Vehicle Routing Problem (V.R.P) 41
2.3 Vehicle scheduling problem (V.S.P) 43
2.4 Vehicle routing and scheduling problems (V.R.S.P) 43
2.5 Convergence with Supply Chain Management (S.C.M) 43
2.6 Related works 44
2.7 Proof of selected algorithms 48
2.8 Research motivation 51
2.9 Conclusion 53
Conclusion (First PART) 54
SECOND PART: MODELLING AND RESEARCH METHODOLOGY 55
Chapter III – A hybrid algorithm Tabu Search – GRASP for wounded evacuation in disaster response. 59
3.1 Introduction 59
3.2 Literature Review 59
3.3 Problem description and mathematical modelling 60
3.3.1 Problem description 60
3.3.2 Assumption 61
3.3.3 Mathematical Formulation 62
3.3.4 Modeling presentation 63
3.4 Prove of NP-Completeness 64
3.5 Resolution methods 65
3.5.1 GRASP 65
3.5.2 Tabu Search 65
3.5.3 Hybrid Algorithm 67
3.5.4 *Experimental results and Discussions* 68
3.6 Failure tests 76
3.7 Conclusion 82

Chapter IV – A hybrid GRASP-Tabu Search approach to solve the integrated problem of ambulance missions' scheduling and resources assignment in the events of multiple disaster zones. 87
4.1 Motivation and Introduction 87
4.2 Mission 88
4.3 Problem description and mathematical modelling 88
4.4 Problem description 89
4.5 Mathematical Formulation 91
4.6 Resolution methods 92
4.6.1 GRASP-TS 92
4.6.2 Hybrid Algorithm 94
4.7 Experimental results and Discussions 96
4.8 Conclusion 105
Conclusion (Second PART) 107
Publications 108
Conclusion 111
Perspectives 112
Bibliography 115

Table de figures

Figure 1. Tabu Search procedure 28
Figure 2. ACO procedure 29
Figure 3. A Star algorithm 30
Figure 4. Greedy algorithm 35
Figure 5. (V.R.P) 42
Figure 6. Distant wounded evacuation 60
Figure 7. Initial Solution 66
Figure 8. Neighbor 1 66
Figure 9. Neighbor 2 66
Figure 10. Neighbor 3 66
Figure 11. GRASP Algorithm model one 67
Figure 12. Hybrid GRASP-TS procedure, Model one 68
Figure 13. CPLEX vs Hybrid Algorithm – comparison of processing time / with capacity constraint. 74
Figure 14.CPLEX vs Hybrid Algorithm – comparison of processing time / without capacity constraint. 74
Figure 15. CPLEX vs Hybrid Algorithm – comparison of solution value / with capacity constraint. 75
Figure 16. CPLEX vs Hybrid Algorithm – comparison of solution value / without capacity constraint. 75
Figure 17. TS VS CPLEX deviation. 80
Figure 18. CPLEX vs. TS; value and time deviation 80
Figure 19. CPLEX vs. TS; value and time deviation - Big dimensions instances 81
Figure 20. CPLEX - Tabu - TS_SA deviation 82
Figure 21. Ambulance-crisis zones- hospital assignment methodology 89
Figure 22. GRASP procedure 95
Figure 23. GRASP_TS procedure 96
Figure 24. CPLEX Vs GRASP-TS, visual results with capacity constraint. 103
Figure 25. CPLEX Vs GRASP-TS CPU consuming time comparison with capacity constraint. 103
Figure 26. CPLEX Vs GRASP-TS values comparison (Capacity constraint excluded) 104
Figure 27.CPLEX Vs GRASP-TS processing time comparison (Capacity constraint excluded) 104

Liste des tableaux

Table 1. Problem characteristics. 70
Table 2. Comparison of obtained results of applied methods with capacity constraint. 71
Table 3. Comparison of obtained results of applied methods without capacity constraint. 73
Table 4.Small size instances. Part I 77
Table 5. Medium-Size Instances 79
Table 6. CPLEX vs TS_SA 81
Table 7. Parameter's values 96
Table 8. Hospitals available capacities 98
Table 9. CPLEX vs C.TS vs GRASP-TS with capacity constraint. 100
Table 10. CPLEX Vs C.TS Vs GRASP-TS, without capacity constraint. 102

Résumé

Les travaux présentés dans cette thèse s'inscrivent dans le cadre des méthodes d'évacuation des populations. Ils visent à étudier les capacités et modéliser le problème d'évacuation (blessés, sinistrés, enfants, personnes âgées, etc.) dans une situation de crise (attentats terroristes, catastrophes naturelles, etc.) et développer des méthodes d'aide à la décision tout en proposant une meilleure planification et des plans optimaux d'évacuation des populations de la zone de crise vers les centres hospitaliers.

L'optimalité est représentée par une valeur maximisant ou minimisant une fonction *f*, dite fonction objective, ou fonction de coût, ou parfois fonction économique.

On se trouve donc face à une chaîne logistique « particulière » qui débute par la demande d'approvisionnement, déclarée par les zones en état de crise et adressée aux zones fournisseurs, et qui se termine par la réception de support et des moyens demandés pour satisfaire cette demande.

Notre approche considère le problème comme une chaîne logistique dans laquelle nous avons plusieurs zones « fournisseurs » et plusieurs zones « terminales ». Dans cette approche, le but fondamental est d'assurer le transport des biens d'une ou de plusieurs zones fournisseurs vers une ou plusieurs zones terminales géographiquement séparées.

Notre travail consiste à résoudre le problème d'évacuation de blessés dans des zones de crise avec une nouvelle vision qui consiste à optimiser le temps de transport en le réduisant au minimum et par conséquence sauver le maximum des personnes touchées par cette crise d'une façon dynamique, efficace et rapide pour minimiser la perte humaine.

Pour achever ce but, trois problèmes se présentent :

- Problème d'affectation des ressources (ambulances et centres hospitaliers) aux blessés.
- Problème d'ordonnancement.
- Problème de capacité maximale à respecter.

Nous proposons deux approches qui permettent de résoudre numériquement ces problèmes.

La première approche suppose que tous les blessés sont dispersés dans plusieurs locations, géographiquement séparées, comme s'est passé dans l'attaque de Paris en 2015.

La deuxième approche suppose que les blessés soient regroupés dans des zones dont chacune possédant un degré d'urgence différent de l'autre. Dans ce cas, tous les blessés appartenant à une même zone sinistrée ont la même distance de leur zone au centre hospitalier.

Enfin, Les deux modèles ont été résolus par une méthode exacte en utilisant le logiciel CPLEX et par un algorithme hybride combinant la méthode de la recherche Tabu et l'algorithme GRASP. Cet algorithme hybride proposé est appelé (TS-GRASP).

Ce système aide l'opérateur humain à être plus efficace à prendre rapidement de bonnes décisions de secours dans des situations d'attentats accidentels, dans le futur, et promet d'être un outil essentiel d'aide à la décision.

Introduction Générale:

) *Contexte général*

Aujourd'hui, les crises naturelles apparaissent avec une fréquence de plus en plus élevée, ce qui provoque des inondations, des tremblements de terre, des incendies considérables, des tsunamis, etc. En plus, des crises artificielles prennent leur place, due à plusieurs facteurs régionaux et internationaux, par exemple, la catastrophe nucléaire de Tchernobyl en 1986, les fuites dans les réacteurs nucléaires japonais à Fukushima en 2011, les attentats terroristes (Tripoli-Liban, Paris-France, Caire-Egypte) …

Plusieurs projets de recherche ont été réalisés afin de développer des outils d'aide à la décision dans des situations de post catastrophe en zone urbaine et de fournir des informations précises en temps réel au centre de gestion et aux équipes de secours. L'ensemble des données collectées partout dans une région sinistrée forme une base de données sur laquelle on peut s'appuyer pour prendre des décisions de secours efficaces, et en même temps rapides. A partir de quelques données, l'opérateur humain peut facilement prendre des décisions efficaces et exactes. Mais avec un très grand nombre de données (c'est le cas dans les situations urgentes), il nous faut avoir un système informatique capable de traiter automatiquement ces données et faire un tri efficace pour récolter les informations importantes pour aider l'agent humain à bien comprendre la situation et prendre rapidement les meilleures décisions de secours.

Généralement, après la déclaration de la crise, un état de désordre s'apparait et les ressources sont généralement mal distribuées.

Les trajectoires suivies par les ambulances dès leurs dépôts jusqu'à leurs destinations sont indéfinies et sujets à plusieurs modifications supplémentaires tout au cour de la crise que ce soit dans le choix des chemins à parcourir, ou dans le choix de leurs destinations (blessés, hôpitaux), ce désordre conduit à une perte de temps et à une mauvaise gestion de ressources valables (ambulances, hôpitaux) ce qui augmente notre perte au plusieurs niveaux surtout au niveau du coût de fonctionnement et au niveau humanitaire (perte de la vie des blessés).

) *Objectif*

L'objectif général de notre travail est de résoudre le problème intégré de transport/allocation de blessés, dans les cas les plus difficiles, pour minimiser au maximum la perte humaine.

Le cœur de notre travail consiste à présenter une stratégie permettant de traiter les problèmes suivants :

1. Servir les blessés selon leurs priorités (ordonnancement selon leurs degrés

d'urgence).

2. Affectation optimale des ambulances aux blessés.
3. Affectation optimale des blessés aux hôpitaux.
4. Optimiser le temps de transport global de toute l'opération (évacuation de tous les blessés).

Pour vérifier et valider notre modèle mathématique, nous l'avons appliqué et testé sur différents scénarios d'affectation des blessés (avec différents nombres et différentes locations), des hôpitaux (avec différentes capacités et locations) et d'ambulances (avec différentes locations et différents nombres) pour montrer son efficacité et sa rapidité de fournir une solution.

Plan synthétique de la thèse.

Notre thèse est décomposée de deux parties.

La première partie a pour objectif de présenter notre revue de littérature qui s'est portée sur les travaux abordant la tournée de véhicules, la chaîne logistique, les différentes stratégies de résolution et le cadre conceptuel de notre recherche. Elle est divisée en deux chapitres :

Le premier chapitre **(chapitre 1)** présente la chaîne logistique et ses différents problèmes de coopération, une description du problème d'optimisation associé et une présentation des méthodes exactes de résolution. Ensuite, dans ce chapitre, on présente d'une façon générale, les méthodes heuristiques et méta-heuristiques :

En premier lieu, nous présentons les algorithmes les plus utilisés pour la résolution des problèmes d'optimisation combinatoire complexes dans une chaîne logistique. En second lieu, nous présentons les algorithmes évolutifs.

Enfin on décrit l'importance et l'efficacité des algorithmes hybrides et leurs applications.

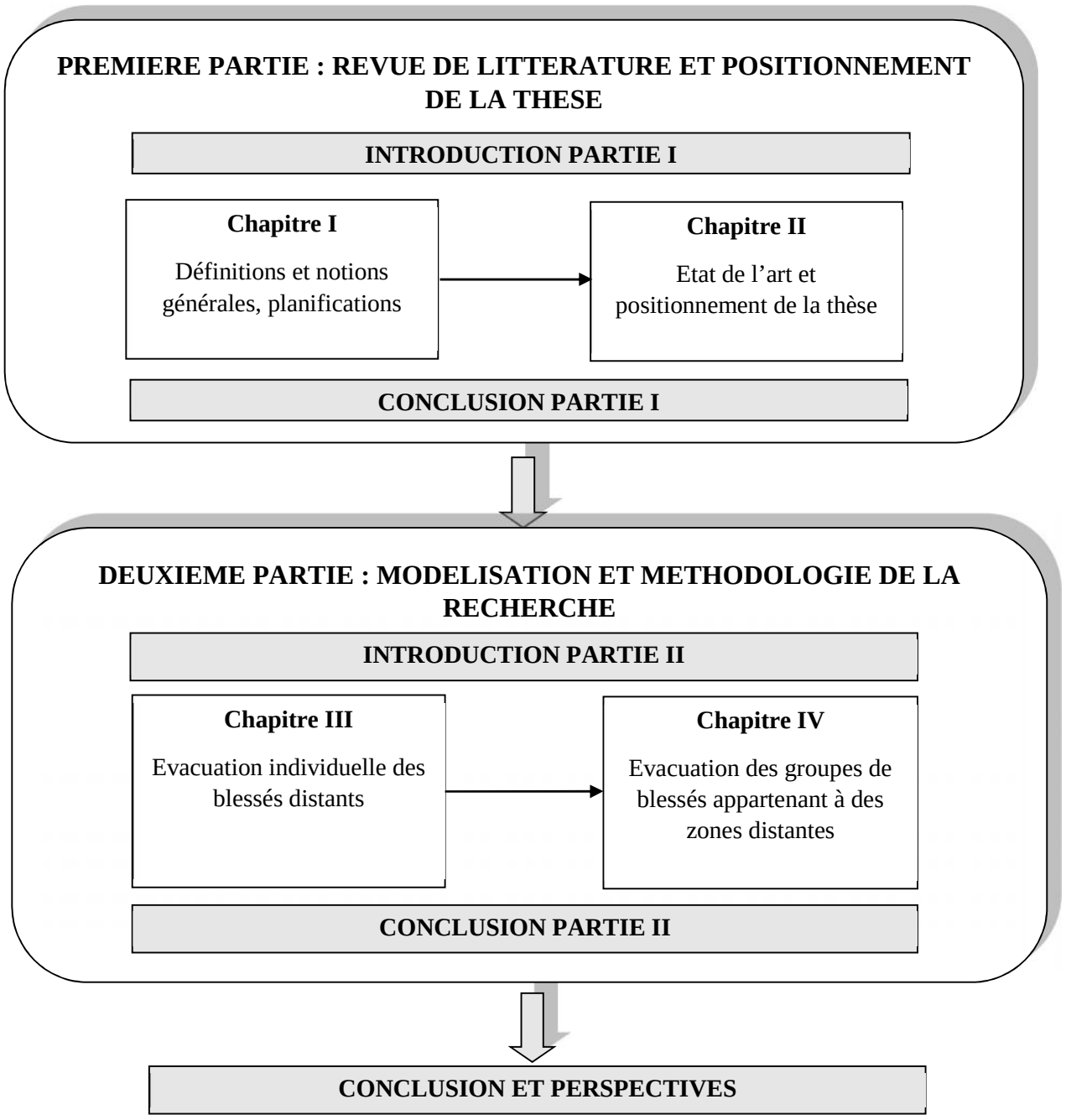
Le deuxième chapitre **(Chapitre 2)** positionne notre problème et les objectifs qui ont motivé nos travaux de recherche. En effet, dans ce chapitre, on met l'accent sur la convergence de notre approche avec la chaîne logistique, ainsi que les apports des algorithmes de recherche basés sur les tournées de véhicules (V.R.P) et (C.V.R.P), et l'ordonnancement des véhicules (V.S.P) par rapport à notre problématique. Enfin, cette étude justifie notre choix de résolution que ce soit au niveau de conception de la méthodologie de développement ou dans l'application de ces méthodes de recherche. Une étude bibliographique dans le domaine de gestion de crise est également présentée.

Dans **la deuxième partie,** nous présentons les résultats obtenus suite à la modélisation mathématique de problème de recherche traité et les méthodes de résolution adaptée.

Dans le premier chapitre de cette partie **(Chapitre 3),** on décrit notre première approche, étape par étape, tout en débutant par la modélisation mathématique du problème puis par la résolution de ce modèle avec une méthode exacte et enfin l'application d'une méthode hybride basée sur la recherche Tabou et l'algorithme (GRASP) (Greedy Randomized Adaptive Search Procedure), appelée (TS-GRASP).

Le second chapitre de cette partie **(Chapitre 4),** présente la deuxième approche qui résout un autre problème de transport dans un scénario différent et avec une conception totalement différente de celle résolue avec la première approche où les blessés à transporter sont regroupés ensemble dans plusieurs zones distantes. Nous développons un nouveau modèle mathématique résolu d'abord par une méthode exacte, puis par une adaptation de la méthode hybride (TS-GRASP).

Le manuscrit se termine par une conclusion générale qui résume les trois approches réalisées et propose, comme perspective, une nouvelle conception qui résout le modèle présenté dans la deuxième approche d'une manière totalement différente tout en utilisant le principe de traitement d'images et l'hybridation entre GRASP et TS.


PREMIERE PARTIE : REVUE DE LITTERATURE ET POSITIONNEMENT DE LA THESE
INTRODUCTION PARTIE I
Chapitre I
Définitions et notions générales, planifications
Chapitre II
Etat de l'art et positionnement de la thèse
CONCLUSION PARTIE I
DEUXIEME PARTIE : MODELISATION ET METHODOLOGIE DE LA RECHERCHE
INTRODUCTION PARTIE II
Chapitre III
Evacuation individuelle des blessés distants
Chapitre IV
Evacuation des groupes de blessés appartenant à des zones distantes
CONCLUSION PARTIE II
CONCLUSION ET PERSPECTIVES

General Introduction

General context

Today, natural crises appear with increasing frequency, which causes floods, earthquakes, considerable fires, tsunamis, etc. In addition, artificial crises take their place, due to several regional and international factors, for example, the nuclear disaster of Chernobyl in 1986, the leaks in the Japanese nuclear reactors in Fukushima in 2011, the terrorist attacks (Paris-France, Tripoli -Lebanon, Cairo-Egypt) Etc.

Several research projects have been conducted to develop decision support tools in post-disaster situations in urban areas and to provide accurate real-time information to the management center and rescue teams. All data collected, everywhere, in a disaster area, forms a database that can be relied on upon to make effective and, at the same time, fast rescue decisions. From some data, the human operator can easily make effective and accurate decisions. But with a very large amount of data (this is the case in urgent situations), we need to have a computer system capable to automatically process this data and efficiently sorts out important information that help the human agent to understand the situation and to quickly make the best relief decisions.

Generally, after the declaration of a crisis, a state of disorder appeared and resources are generally poorly distributed.

The trajectories followed by ambulances from their depots to their destinations are indefinite and subject to several additional changes during the crisis, whether in the choice of paths to be covered, or in the choice of their destinations (wounded, hospitals), this disorder leads to a waste of time and poor management of valuable resources (ambulances, hospitals) which increases our loss at several levels especially in terms of operating costs and humanitarian level (loss of life of the wounded) .

Goal

The general objective of our work is to solve the integrated problem of transport / allocation of wounded, in the most difficult cases, to minimize the human loss.

The heart of our work is to present a strategy to address the following issues:

1. Serve wounded according to their priorities (scheduling according to their degrees of urgency).
2. Optimal assignment of ambulances to the wounded.
3. Optimal allocation of wounded to hospitals.

4. Optimize the overall transport time of the entire operation (evacuation of all the injured).

To verify and validate our mathematical model, we applied and tested it on different scenarios of wounded assignment (with different numbers and different locations), hospitals assignment (with different capacities and locations) and ambulances (with different locations and different numbers) to show its effectiveness and speed in providing solution.

Thesis Synthetic plan.

Our thesis is divided into two parts.

The purpose of the first part is to present our literature review on the vehicle routing problem, the supply chain, the various resolution strategies and on the conceptual framework of our research. It is divided into two chapters:

The first chapter **(chapter 1)** presents: the supply chain and its various cooperation problems, a description of the associated optimization problem and a presentation of the exact methods of resolution. Then, in this chapter, we present, in a general way, the heuristic and meta-heuristic methods:

First, we present the most used algorithms for solving complex combinatorial optimization problems in a supply chain. Secondly, we present the evolutionary algorithms and finally, the importance and effectiveness of hybrid algorithms and their applications are described.

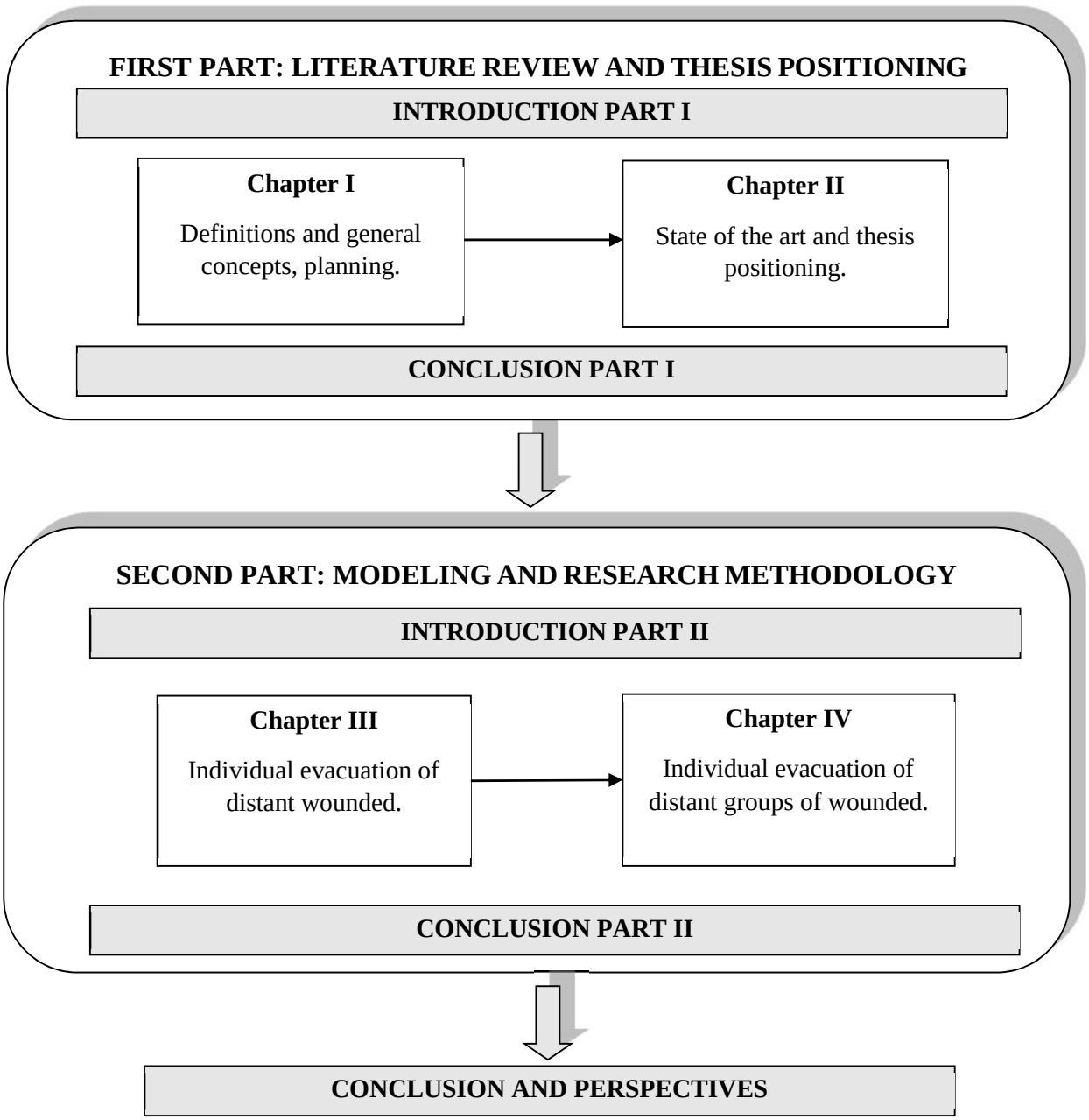
The second chapter **(Chapter 2)** positions our problem and the objectives that motivated our research. Indeed, in this chapter, we focus on the convergence of our approach with the supply chain, as well as the contributions of search algorithms based on (VRP), (CVRP), and (VSP) in relation with our problem. Finally, this study justifies our choice of resolution methods whether it is at the development methodology conceptual level or in the application of these research methods. A bibliographic study in the field of crisis management is also presented.

In the second part, we present the mathematical modeling of our treated research problem, the resolution methods and the obtained results.

In the first chapter of this part **(Chapter 3)**, we describe our first approach, step by step, while beginning with the problem mathematical modeling continuing with the resolution of this model with an exact method, and ending with the application of a hybrid method based on the sequential hybridization of Tabu Search and the (Greedy Randomized Adaptive Search Procedure) (GRASP) algorithm, called (TS-GRASP).

The second chapter of this part **(Chapter 4)**, presents the second approach that solves another optimization problem in a different scenario and with a totally different concept where wounded, subject to rescue, are grouped together in several distant zones. We develop a new mathematical model that is solved by an exact method, and then by an adaptation of the hybrid method (GRASP-TS).

The manuscript ends with a general conclusion that summarizes the three approaches and proposes, as a perspective, a new conception that solves the model presented in the second approach in a totally different way while using the image processing principle and the Hybridization between GRASP and TS.


FIRST PART: LITERATURE REVIEW AND THESIS POSITIONING
INTRODUCTION PART I
Chapter I
Definitions and general concepts, planning.
Chapter II
State of the art and thesis positioning.
CONCLUSION PART I
SECOND PART: MODELING AND RESEARCH METHODOLOGY
INTRODUCTION PART II
Chapter III
Individual evacuation of distant wounded.
Chapter IV
Individual evacuation of distant groups of wounded.
CONCLUSION PART II
CONCLUSION AND PERSPECTIVES

FIRST PART: LITTERATURE REVUE AND THESIS POSITIONNING

This part aims to present the theoretical research framework.

We will first present the definition of logistics, most recognized transport problems, supply chain and some of the existing resolutions methods.

The first chapter represents a bibliographic study and a state of the art of the different optimizations methods and exact resolutions methods used to solve complex combinatorial problems.

The second chapter is devoted to the presentation of the various applicable resolution strategies during sudden crises and to the explanation of the most adopted algorithms until now for solving transport problems within a crisis management supply chain. We will also highlight the relationship between the approximate methods and their applications to our area of research, its benefits and its risks.

Chapter I

Preparation and general definitions

Chapter I – Preparation and general definitions

1.1 Introduction

Since the beginning of the 20th century, the technological advancement at all levels has led to a huge evolution in several sectors, especially the information systems. In this research we will try to benefit from this development for the sake of life saving. During a crisis, we aim to reduce the loss to the minimum and hope to return to the prior crisis state as much as possible. The significant development of this type of computerization requires the collaboration and synchronization between all the actors and all the interdependent tasks in the work area to achieve a global benefit.

Supply chain integration tends to standardize and organize the routing of interrelated tasks.

In the context of a supply chain, the work plan must define its objectives and the foreseen contradictions.

This chapter presents the context of our study and its relationship with the crisis management supply chain and approximate resolution methods.

1.2 Logistics and supply chain

In 2001, according to Akbari "etymologically, the term logistics comes from the Greek". The definition of the supply chain has evolved over time. In the eighteenth century, France was the first nation that, with the emergence of problems of continuous supply of military resources (weapons, support, food, ...), used logistics. After several years this term is widespread in the production section especially in the industrial circles within companies, (Colin, 1981), have proposed the following definition: Logistics is the strategic process by which the company organizes and supports his activity. As such, we can determine and manage the related material and information flows, internal and external, upstream and downstream.

Several types of logistics are available: supply logistics, production logistics, distribution logistics, military logistics and finally retro-logistics.

1.2.1 Transport logistics

Logistic and transport are closely linked, we can have different types of transport operations such as commodity transportation, passenger transportation, relocation etc.

For better exploitation, we can distinguish two sides of information systems:

I. Regulator side (operators):

The operation goes through two phases:

a. Planning: It involves designing, planning and scheduling transportation resources
b. Regulation: It consists in refining the schedules and the number of vehicles at peak hours to improve the quality of service and optimize the use of the network.

II. Client side (traveler):

It is very similar to Traveler Salesman Problem (T.S.P) where we have interest in reduction of both costs and traveling time.

At the transport problem level, the supply chain application relies on two transport systems:

A. Humans transportation system:

It depends on:

a. Transporting tools
   - ✓ Infrastructure (transport network).
   - ✓ Equipment (Train, bus, Tram and cars).
b. Operators who invest and manage the network.
c. The passengers who use this network.

B. Commodities transportation system:

They consist in making available to the customer, the quality products in quantity and in the promised time and at an economic and competitive cost, such system includes:

a. Choice of transportation mode: Roads, air, combined.
b. Choice between available, outsourced or hybrid transportation.
c. Packing type.

d. Stock management.
e. After sale services.

1.2.2 Crisis management supply chain

Crisis have several forms they could be natural (Tsunami, floods, earthquake etc.) or triggered (war, nuclear accident, terrorist attacks etc.). Crisis evolvements could be either quick or slow depending on crisis type and urgency degree.

The fast reaction is compulsory in order to deal well with its consequences and return as fast as possible to the previous state before crisis announcement; a good methodology should be adopted for an efficient crisis management.

Crisis management includes:

Crisis prevention: Some crises can be predictable such as drought; famine ... The objective of this step is to prevent the crisis from actually happening.

Operational response competence: It is the speed of the reflex (mobility, deployment of the necessary resources...) these reflexes must be planned in advance.

Task management: It concerns all the various synchronized and well-sequenced operations that are set in motion during the crisis with the aim of restoring the situation by minimizing its effects.

1.2.3 Supply chain

The supply chain is defined as the network of companies that participate, upstream and downstream, in the various processes and activities that create value in the form of products and services provided to the end consumer (Christopher, Logistics and supply chain managment, 1992). Thus, it can be defined as a network of facilities that provides the functions of supplying raw materials, transforming these raw materials into components and finished products, and distributing the finished product to the customer (Billington, 1993). Then after a few years (R. R. Lummus, Strategic supply chain planning, 1998) defined it as a network of entities through which the material flow passes; these entities include suppliers, carriers, assembly site, distribution center, retailers and customers.

In fact, it brings together approaches, processes and functions that are essential for reducing costs and increasing its flexibility in order to optimize its performance, in other term, improve the overall production system and provide effective decision support.

At a company, and whatever its domain of activity was, three levels of decision-making are presented:

- **Strategic level:** Set of guideline-type decisions making and long-term action lines "6 months or more" (Croom, 2000).
- **Tactical level:** A set of medium-term decisions (from a few weeks to a few months) dealing with the issues related to the management of company resources available over a fixed horizon.
- **Operational Level:** Set of daily and weekly decisions generating a detailed production or scheduling plan, (J. Thomas, 1996).

The humanitarian supply chain has the same objective as companies supply chain but, in a context of great disruption, it is more difficult to set up, its characteristics are identified in five categories:

- The life cycle of the humanitarian operation.
- Environnement.
- The nature of flow.
  - ✓ Physical flows.
  - ✓ Information flows.
- Logistic network
- The Dynamics
  - ✓ Evaluate urgent and essential needs that might be uncertain.
  - ✓ Coordinate the supply chain
  - ✓ Work in emergency conditions

Take into consideration that the provided data are uncertain and subject to some subsequent modifications resulting from lack of transparency.

Modeling:

Analytical model: Set of mathematical equations to describe the system, note that it is necessary to make a certain number of assumptions and reductions for the analytical model to be simple (deterministic or stochastic model), analytical models are generally content with aspects based on distribution in a dyadic structure (Huang, The impacts of sharing production information on supply chain dynamics: a review of the literature, 2003).

Simulation model: This model is used when there is no relationship between the different variables of the system. It is both stochastic and dynamic; the different

simulation techniques of logistic chains are classified according to (Maria, 1997), in three families:

a. Spreadsheet simulation,
b. The systems dynamics approach,
c. The discrete events approach, which educate and train users in certain aspects of supply chain management.

1.2.4 Supply chain structure

In 2001, Beamon and Chen proposed a structuring of the supply chain divided into four families "convergent, divergent, joint and general".

- The convergent chain, the materials coming from several supplier sites converge towards a single destination site, which assembles them together like the case of the car building industries.
- The divergent chain is the opposite of the convergent chain where materials start from a single source and share through the chain. For example, the mining industry.
- The joint chain is the union of a convergent chain and a divergent chain.
- General chains are neither totally convergent nor totally divergent.

1.2.5 Control

Controlling is a function that exploits the available resources of a multi-entity supply chain.

This function permits storing, adapting and making available to use all of the data that decision centers count on to provide efficient decision. We can recognize two types of control.

1.2.5.1 Centralized control, conventional centralized systems:

a. *Manufacturing Resource Planning systems (MRP):* Their main goal is to provide a decision support tool that uses an efficient planning method for all resources of an industrial enterprise.

   The components of this method are:

   - Strategic plan.
   - Industrial and commercial plan.

- Production Manager Program.
- Calculation of needs.
- Planning of capacity needs and monitoring the execution of plans.

b. ***Enterprise Resource Planning systems (ERP):*** They are based on MRP and cover several functions, among them:

- Reliable access to information.
- Centralization of data and simultaneous access.
- Proposal of E-Commerce functions.
- Some additional modules.

c. ***Advanced Planning System (APS):*** Their main goal is to obtain optimized solutions at different decision levels (Herrera, 2011) while using different techniques and algorithms.

In addition to (ERP), (APS) provides a solution that optimizes collaborative work according to predefined objectives.

The main components are :

- Production planning.
- Production scheduling.
- Distribution and transportation planning.

1.2.5.2 Evaluation extent in a state of crisis

Evaluation is an essential element for provisioning estimated needs in the disaster area. Afterwards, we can specify the most urgent cases to make intervention, the quality of this evaluation is very important, because all following intervention operations will be heavily dependent on the pre-identified situation in the operations field.

Good assessments must consider: wounded needs in the affected areas, resource needs and their local availability, the infrastructure capacity and finally respect the obligations.

1.3 Optimization problem

Optimization can be either a maximizing or minimizing value of an objective function *f*.

It is a question of finding among the set of decision variables, the element that optimizes the optimization criterion.

Two essential aspects bring optimization methods together:

- Distributed calculation.
- Problems of adaptation and research self-adaptation.

(Ejday, 2011) defines the concept of optimization as comprising two main phases:

- Modeling.
- Problem resolution.

The efficient optimization method should consider the following points:

- ***Robustness:*** Standard method applicable for a wide class of problems (ability to adapt to datasets variations).
- ***Efficacy:*** It should not be expensive in CPU time and RAM.
- ***Precision:*** The numerical result must hang over the real problem optimized solution.

The complexity of a decision problem is measured against the lower complexity of the algorithm that solves this problem.

Let us illustrate some complexity classes:

- ***The P class:*** A problem P is considered as polynomial if it can be solved with an algorithm of polynomial complexity, all of these problems form the class P.
- ***The P-Space class:*** We are resorted to this problem if exist an algorithm that solves a problem P in polynomial space with respect to the data size.
- ***The class NP:*** If we have for every instance of a problem P a solution of the problem in polynomial time, we say that P is in NP.
- ***The NP-Complete Class:*** Notion of dominance between problems, P dominates P' if we can transform any instance I of P into an instance I' of P' due to a polynomial algorithm. I is a solution of P if and only if I' is a solution of P'.

1.4 Mathematical modeling encountered problems

1.4.1 Linear model

It is a mathematical model that represents an optimization problem. The most popular method is the simplex method, which is a very efficient method in practice, but the calculation time can become very large, that is why we are forced to use additional techniques (ex. Branch and Cut) to minimize the overall processing time.

1.4.2 Binary linear model

In binary linear modeling, many problems have encountered such as assignment problem, maximum flow problem, and integer production problem.

For some problems where we look for an entire optimal solution (ex. whole quantity of product…), we must tackle the problem with integer constraint in the program; otherwise the optimal solution is not complete (backpack problem, filling problem).

1.4.3 General principle of cutting methods

In general, these methods are repetitive, at each repetition:

a) We ignore the integrity constraint on the variables (simplex) to solve the linear program and consequently we obtain an optimal solution that can be complete.
b) A complementary constraint is added to remove only the non-complete solution already obtained. This additional constraint is called a cut.
c) Repeat the previous steps from the beginning in each repetition until we obtain a complete solution.

1.5 Exact construction methods

1.5.1 Branch & Bound (B&B)

The principle is to generate the tree progressively by trying to start with the most promising sub-trees and especially by not exploring to the leaves of a sub-tree, which we know that it does not contain the optimal solution.

This method is applied for solving combinatorial optimization problems with a large number of possible solutions. We can associate this method with a tree whose root contains all of the feasible solutions. The other vertices emanating from the root are governed by the B&B method (Doig, 1960), which is based on a tree search method for an optimal solution by separations and evaluations, representing the solutions states by a state tree, with nodes, and leaves.
The B&B algorithm is based on three main axes:

a. *The separation (Branch):*

The separation aims to subdivide the problem into sub problems. The root of the tree represents all of the feasible solutions. Thus, by dealing with all sub-problems and keeping at each research step the best solution found, the initial problem is

definitely solved. This is equivalent to establish a tree that permit counting the set of all solutions. The remaining undiscoverable set of tree nodes that can contain an optimal solution is called set of active nodes.

b. *The evaluation (Bound):*

The fundamental purpose of evaluation is to simplify the search space while eliminating some subsets that do not contain the optimal solution. The goal is to evaluate the exploration interest of the tree subset. The branch & bound eliminates branches in the search tree as follows:

The search for a minimum cost solution requires memorizing the lowest encountered cost solution during exploration phase and then comparing the cost of each traveled node with the best solution cost. If the cost of the considered node is higher than the best cost, we stop exploring the branch and all of its solutions that will necessarily be with higher cost than the best solution cost already found.

c. The course strategy:

- Width: This strategy privileges the root closest vertices by making fewer separations from the original problem. It is less effective than the other two strategies presented.
- Depth: This strategy favors the farthest vertices from the root by applying more separations to the initial problem. This route quickly leads to an optimal solution by saving the memory.
- The best: This strategy consists of visiting sub-problems with the best terminal. It also avoids the exploration of all the sub-problems that have a bad evaluation compared to the optimal value.

1.5.2 Branch and Cut

This method represents a very powerful resolution algorithm for different linear or binary linear model and it can guarantee the optimality, it is an exact method, which consists in the combination between the cutting plane method and the branch-and-bound algorithm.

We find that:

Branch & Cut = Branch & Bound + Cutting Planes.

Generating Cutting Plan: The general idea is to add constraints that eliminate the fractional solutions of linear programming without eliminating the entire solutions, which, in the first place, greatly reduces the size of the Branch-and-Bound tree and in second place, speed up execution. The Branch-and-Cut algorithm modifies the Branch-and-Bound's basic strategy by attempting to enhance the relaxation of the linear programming of an IP (Integer Programming) with new inequalities before connecting a partial solution.
Branch & Cut can be used in conjunction with heuristic methods to obtain a lower bound of the optimal value while using the Branch & Bound algorithm. This method is very popular.

1.5.3 Simplex method

It is a repetitive method that aims to start with a basic solution (feasible but not optimal), then to improve it at each iteration in order to obtain the optimal solution, at the base it builds tables where the will be manually calculated.

In the first place, the simplex algorithm (Dantzig, 1987) models the problem in the form of an objective function and constraints (canonical program), in the second place it transforms the canonical program into a standard program by introducing gap and artificial variables to turn inequality into equations.

Algorithm Principles:

- ✓ Start from an extreme point.
- ✓ Move (pivot) to a nearby extreme point improving the solution.
- ✓ Repeat until the optimum solution is obtained.

1.5.4 Dynamic programming

Dynamic programming is a programming paradigm, that is, a particular way of understanding a given algorithmic problem.

It is a useful method for obtaining an exact solution to an algorithmic problem, where a "classical" solution is found to be too complex and inefficient. This is called combinatorial optimization.

The effectiveness of this method is based on the principle of optimality stated by the mathematician Richard Bellman: "any optimal policy is composed of optimal sub-policies".

1.6 Heuristics and meta-heuristics search

Artificial intelligence (AI) has now become applicable in several areas, such as image processing, robotics, logistics ... but as for the resolution of logistics problems, the AI faced several obstacles especially in the field of solving combinatorial problems where we are obliged to find all the alternatives solutions and to invest them, which is practically impossible. In such situations, powerful search algorithms are used to improve the efficiency of the search.

The heuristic and meta-heuristic methods aim to solve these complex problems, and they are considered as important suppliers of effective AI research methods.

a. Heuristic methods definition:

It is a discipline that proposes to formulate the rules of scientific research. Sometimes it leads to an astonishing result or nothing; it seeks a good solution close to optimality with a reasonable calculation time. The heuristic algorithm solves a given optimization problem but without any guarantee of optimality with a reasonable calculation time ex.: the greedy search algorithm.

b. Meta-heuristic methods definition:

These stochastic optimization algorithms combine several heuristic approaches whose main tasks are the ability to extract from a local minimum to a global minimum. A meta-heuristic can be applied to different combinatorial optimization problems using independent strategies that guide the search in solution space, so as not to remain in local minima ex.: ant colony optimization, Tabu Search, Greedy Randomized Adaptive Search Procedure, Simulated Annealing etc.

1.6.1 Simulated Annealing

The idea is to perform a movement according to a distribution probability, which depends on the quality of different neighbors, best neighbors have a higher probability, and less good ones have a lower probability (Devadas & Newton, 1989).

The search method requires the use of a parameter called temperature and noted *T*, which can vary during algorithm execution process:

- T High: In the solution space, all neighbors have almost the same probability of being accepted; the number of possible choices is very large.

- T low: In the search space, the number of possible choices of neighbors is reduced to the minimum possible. Therefore, we have a low probability of choosing a movement.
- T = 0: We can no longer make any movement.

The temperature varies during the search: at the beginning, T is high, and then it decreases and ends up in tending to zero.

The function that specifies the evolution of the temperature is called the cooling scheme.

In standard simulated annealing, the temperature decreases by level.

Simulated annealing parameters can be adjusted by observing the percentage of acceptance during the search.

If the initial temperature is too high, the start of the search is useless.

- Initial temperature: A parameter sets the initial acceptance percentage,
- Length of a step: A parameter limits the number of tests (iterations) by level, another parameter limits the number of changes (movements) for the beginning of the search,
- Stop criterion: A counter is used to determine if the algorithm stops its progression. The counter is zero at the beginning and increments at each iteration. The search stops when the counter reaches a certain threshold.

At the end of a landing, the counter is:

- Incremented, if the acceptance percentage is below a threshold,
- Reset, if the quality of the best solution has changed during the stage.

1.6.2 Tabu Search

In (Glover, Tabu search–Part I, 1989), (Glover, Tabu search–Part II, 1990), was the first who develops a meta-heuristic search called Tabu Search, which became now widely used for solving complex combinatorial optimization problems of large or small size (often NP- hard). In addition, Tabu Search (TS) has several applications in nonlinear programming (NLP).

This research technique is based on two phases:

a. Adaptive memory that depends on four important references (occurrence, recently visited, quality and influence).
b. Reactive exploration, it is able to take tactical selections to achieve efficiency.

- *Definition:*

  The basic idea of the Tabu list requires the memorization of old visited modifications or regions and to introduce certain mechanisms to prohibit the search to return too quickly to these changes, these mechanisms are temporary prohibitions of certain movements (what is called taboo movements), it is a question of prohibiting the movements which would try to cancel the effect of last executed movements. At each iteration, TS chooses the best neighbor, which does not exist in the Tabu list, even if it can degrade the cost function, for this reason we say that TS is an aggressive method (Figure 1).

- *Memory:*

  The main role of using adaptive memory is to memorize already visited candidates, this memory is called Tabu list, the main purpose of this memory is to restrict the choice of redundant candidates who will be marked as active Tabu for a certain moment; when the size of the list exceeds the user predefined limit, the first selected candidate will be eliminated from the list, this method is called the aspiration. The fundamental purpose of this memory is to avoid the algorithm to enter a local optimum which improves the research, its role evolves during the resolution by allowing us to use the strategies of intensification and diversification, the Intensification strategy allows the algorithm to exploit in addition the good candidates already found, while the diversification strategy allows us to explore in space candidates who are never already visited.

- *Exception to prohibitions*

  It is possible to violate a prohibition when a prohibited movement makes it possible to obtain the best solution registered so far.

- *Specifying the Tabu list mechanism*

  To describe the Tabu mechanism it is necessary to specify:

  - The Tabu list structure (the nature of the elements that compose it).
  - How the list is updated when a move is made, (Which items are added).
  - Which movements or neighbors are prohibited according to the contents of the list.
  - It is also necessary to determine how long an element remains Tabu.

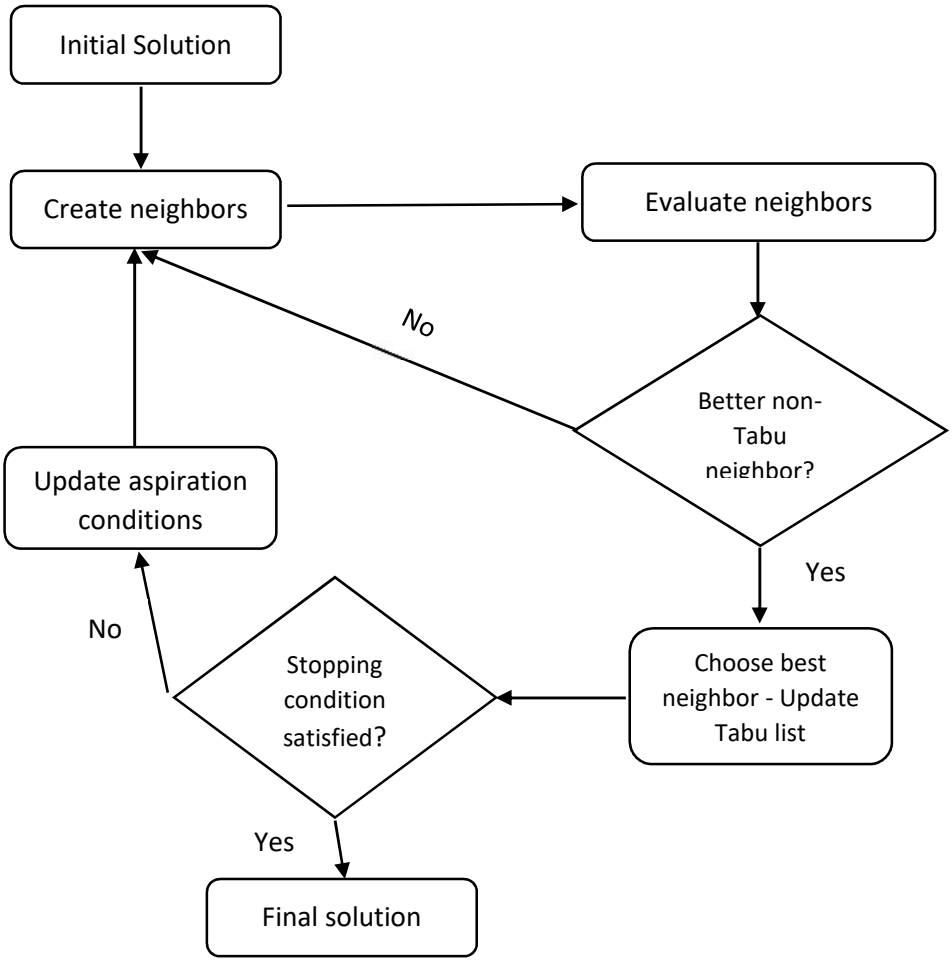


Figure 1. Tabu Search procedure

1.6.3 *Ant colony optimization*

The ant colony optimization is a probability resolution technique; (Dorigo M. , 1992) developed it. The idea is to represent the problem to solve in the form of the search for a better path in a graph based on the actual search method of a path between the colony and the food source.

The behavior of artificial ants is inspired by real ants; actually, ants are blind, that's why they move in a random way and leave traces of pheromone on the roads they cross until reaching food source, more pheromones will be deposited on the road, the more the probability of following this pheromone increases (figure 2). Artificial ants follow the same method but they deposit traces of pheromone on the components of the graph and they choose their paths relative to traces of previously deposited pheromones, these traces evaporate over time.

This meta-heuristic has solved various combinatorial optimization problems, such as the Traveler salesman problem (TSP) (Dorigo M. a., 1997), the quadratic assignment problem and the vehicle routing problem [BUL 99].

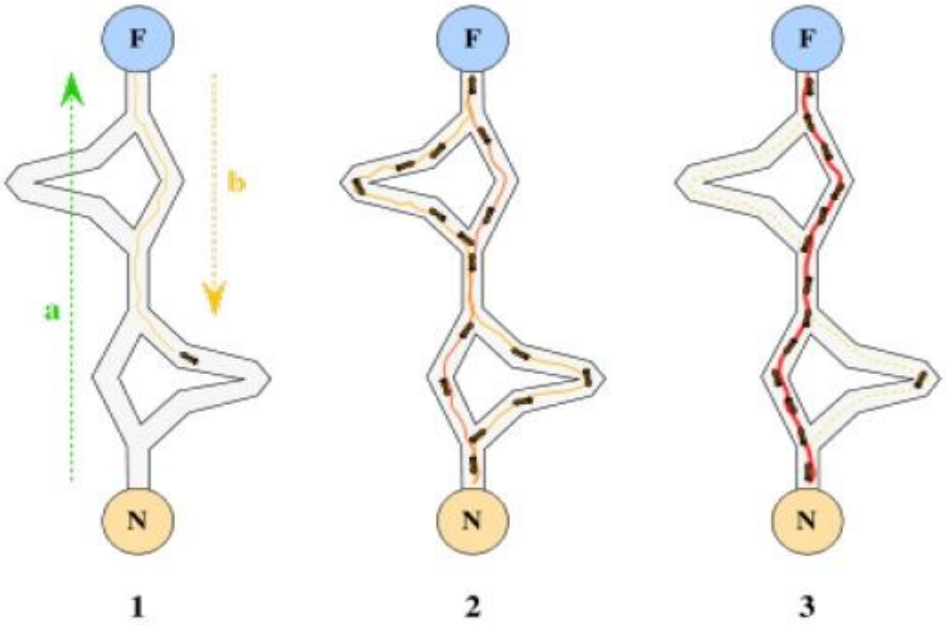


Figure 2. ACO procedure[i]

An ant will move from node i to node j with probability.

$$P_{i,j} = \frac{T_{i,j}^{\alpha} * n_{i,j}^{\beta}}{\sum T_{i,j}^{\alpha} * n_{i,j}^{\beta}}$$

Where:

$T_{i,j}$ is the amount of pheromone on the ridge i; j.

α is the influence control parameter of $T_{i,j}$

$n_{i,j}$ is the desirability of the ridge $i; j$ (usually $\frac{1}{a_{i,j}}$).

β is the influence control parameter of $n_{i,j}$.

The amount of the pheromone is updated according to the following equation:

$$T_{i,j} = (1 - p)\, T_{i,j} + \Delta T_{i,j}$$

Where p is the evaporation rate of the pheromone and Δ is the total amount of the pheromone deposited.

$\Delta T_{i,j}^{k} = \frac{1}{L_k}$ if the ant k passes on the ridge i,j;

= 0 if else;

Where L_k represents the cost of passage of the ant k.

1.6.4 Scalable Algorithms

1.6.4.1 A* Algorithm

It is an algorithm for finding a shorter path in a graph from x0 to x_f.
Consider a two-dimensional matrix (figure 3) where we have several obstacles and we will start from any point to another point (destination), we must arrive (if possible) to this destination, the algorithm A * solves this problem as follow:

- Make a heuristic assessment at each vertex to estimate the best-used path.
- The vertices are traversed according to this heuristic evaluation: we retain the summit where the evaluation is the smallest.

In the algorithm A *, we reduce the set of vertices to explore.
At each iteration, it takes a cell whose cost is the smallest and goes to another point by following the same method until arriving at the destination if possible.
The movements of this algorithm depend on the cost of passage.

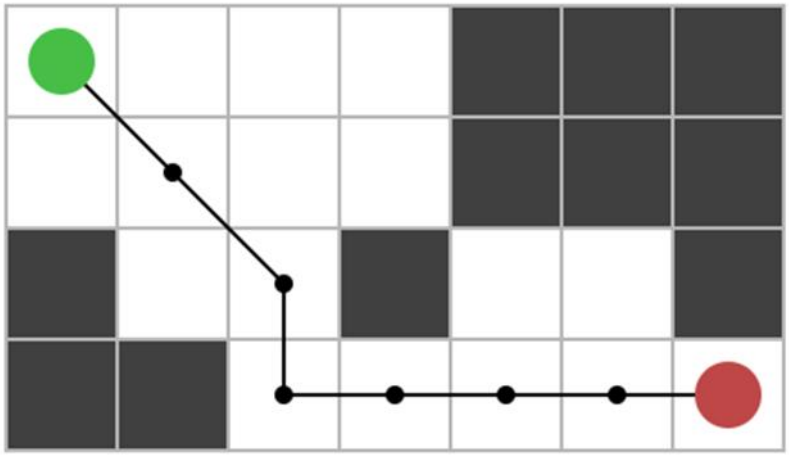


Figure 3. A Star algorithm

1.6.4.2 Genetic Algorithms

Genetic algorithms are randomized and intelligent adaptive exploration metaheuristic algorithms, based on the evolutionary ideas of natural selection and genetics; they are used to solve optimization problems. Their technique consists of simulating the evolution processes of natural systems, to form an exploration algorithm that has some of the human exploration characteristics. At each generation, a new set of artificial creatures (strings) is created using innovative parts. Genetic algorithms are not totally random, they effectively leverage the

information already obtained to speculate on the position of new points to explore, with the hope of improving performance.

The genetic algorithms were developed by John Holland, (Holland, 1975), his colleagues, and his students, at the University of Michigan.

The mechanisms of a basic genetic algorithm are surprisingly simple and involve nothing more complicated than copies of strings and exchanges of strings; it is the reasons why this simple process works is much more complex and subtler. Easy implementation and efficiency are two of the most attractive features of genetic algorithms.

Advantage:

a. Ability to apply on a wide range of applications without worrying about their nature.
b. Simultaneous treatment on an *n* size population.
c. Using crossover and mutation methods, which emerge towards a global optimum by avoiding local optimums.

Operations :

a. Generate initial population of fixed size *N* individuals,
b. Calculate the values of their objective function,
c. Use a method of selecting individuals,
d. Apply the crossover operator with a probability P_x to generate individuals,
e. Apply the mutation operator on the results already obtained with a probability of P_m mutation to generate individuals,
f. Use a method of insertion of individuals from the new population.
g. Evaluate the value of the objective function of the individuals already inserted in the new population,
h. Quality control of the individuals obtained
    - Good quality => Stop test.
    - Bad quality => Iterate again for the new generation.

Selections Methods:

- Rank selection.
- Roulette selection.
- Random selection.
- Tournament selection.

Rank selection:

Each individual is assigned his ranking in order of

- ✓ *Maximization problem:* Sorting individuals in ascending order depending on the objective function values.
- ✓ *Minimization problem:* Sorting individuals in descending order depending on the objective function values.

Roulette selection:

- ✓ *Maximization problem:* We associate to each individual a probability of selection P_i proportional to its value F_i in the objective function.
- ✓ *Minimization problem:* We associate to each individual a probability of selection equal to $\frac{(1-P\quad)}{(N-1)}$.

Random selection:

Each individual is selected regardless of his or her adaptation value with a uniform probability $\left(\frac{1}{N}\right)$ of being selected.

Tournament Selection:

This method increases the participation possibility of poor-quality individual's, according to their fitness, to the improvement of the population. In an M size subpopulation, poor quality individuals are competing to choose the best quality one with respect of ($M < N$). The winning individual is selected for the application of the crossover operator.

Crossover operators:

Goal: Enrich the diversity of the population by manipulating the components of chromosomes.

Behavior: Let P_x be the crossing probability; Two individuals are already chosen, we generate a random number $\alpha \in [0, 1]$ if $\alpha <= P_x$ so the operator applies to the couple.

Operation on two chromosomes with binary coding:

a) Single point cross operator: Divide each parent into two parts at the same position (randomly selected).
   - Child number 1 is composed of the 1st part of parent 1 and the 2nd part of parent 2.

- Child number 2 is composed of the 1st part of parent 2 and the 2nd part of parent 1.

b) Two-point crossover operator: Divide each parent into three parts at the same position (randomly selected)
   - Child 1 will be the copy of parent 1 by replacing his part between the two positions by that of parent 2.
   - Child 1 will be the copy of parent 2 by replacing his part between the two positions by that of parent 1.

Mutation operator:

This operator guarantees that genetic algorithm will be able to reach most points of the realizable domain.

Consider P_m as mutation probability. Only in genetic algorithms with binary coding, this probability was carried out on the genes by changing its value from 0 to 1 or from 1 to 0 and not on the entire chromosome (the choice of genes to change is random).

With the different coding, genetic algorithms P_m is applied to the whole individual (chromosome) and not to the genes.

Insertion method:

After applying crossover and mutation operators, this method is used to generate a new population.

Three generations strategies are known:

Strategy 1: It consists of choosing N individuals from the N_f children already created by supposing that $N_f \geq N$ if $N_f = N$. It is a generational method that replaces the parent chromosome with a child chromosome.

Strategy 2: It consists of choosing N individuals from N parents of the previous population and N_f new children, but this strategy has a special case called the state of equilibrium method that saves a large part of the population in the next generation and selects the best chromosomes at each iteration to generate child chromosomes. To copy the worst parents, the rest of unselected chromosomes will be copied into the new generation.

Strategy 3: Called elitism, it consists in copying some better chromosomes in the new population to avoid the loss of the best chromosomes after the application of the crossing and mutation operators, this method is very efficient because it improves the algorithms genetic, because it retains at

each iteration the best individual found in all populations previously generated.

Stop test:

In genetic algorithm, the purpose of the stop test is to ensure the optimality or the best-known solution after a finite number of iterations.

Two stopping criteria are known:

- Stop at a prefixed generation number.
- Stop at the time when the population stops evolving or no longer evolves sufficiently.

Note that we cannot ensure the good algorithm convergence due to the stop criterion which is totally arbitrary and the returned solution is only an approximation of the optimum.

Size of *M* is critical because if *M = N*, it means that the provided results of the selection gives each time the best individual with respect to the value of the objective function, so the search becomes single task and gives only one solution at a time, this type of algorithm converges sometimes quickly to a local optimum.

If *M = 1,* then this selection method will be random.

Each individual "*i"* is then reproduced according to its probability P_i which means that the good candidates are the most reproduced and the bad ones will be eliminated.

1.6.4.3 Greedy algorithms

These are the simplest and fastest algorithms because they build a solution iteratively without resorting to the decisions made in previous iterations, the generated solution relies on the best choice taken at each moment, in case algorithm fails to systematically provide the optimal solution, it is considered as a greedy heuristic.
Among the greedy algorithms is the most extensive algorithm, Hill Climbing[ii] (figure 4).

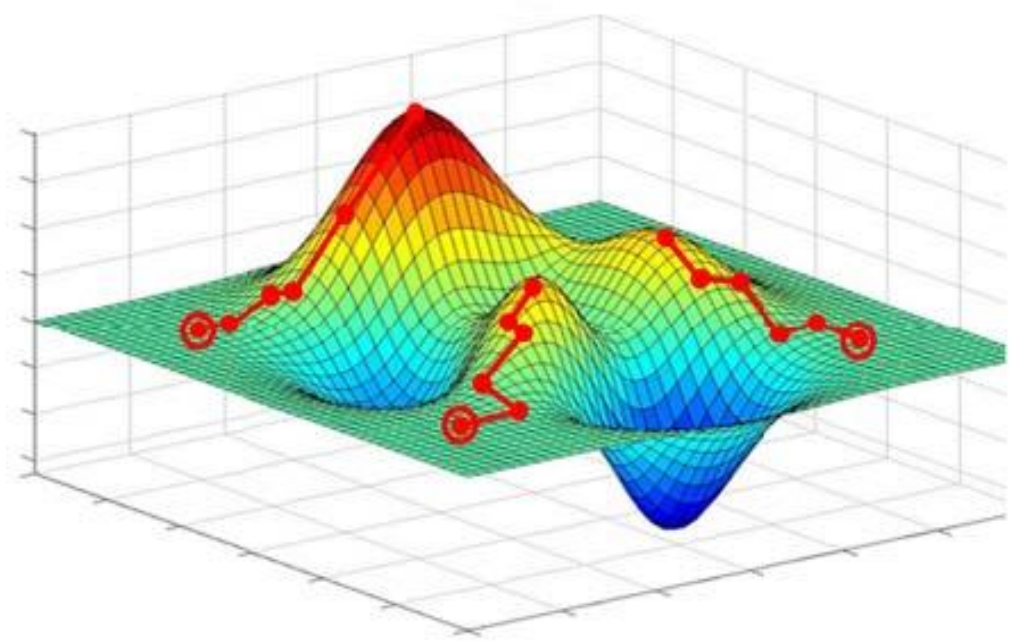


Figure 4. Greedy algorithm

Research algorithm

```
currentNode = startNode;
  loop do
        L = NEIGHBORS(currentNode);
        nextEval = -INF;
        nextNode = NULL;
        for all x in L
              if (EVAL(x) > nextEval)
                    nextNode = x;
                    nextEval = EVAL(x);
              end if
        end for
        if nextEval <= EVAL(currentNode)
              //Return current node since no better neighbors
              exist
              return currentNode;
        end if
        currentNode = nextNode;
  End do
```

1.6.4.4 Greedy Randomized Adaptive Search Procedure (GRASP)

(Feo, 1995) invented the (GRASP) metaheuristic algorithm, it is an iterative search technique, at each iteration, a new feasible solution is constructed using the randomized greedy search procedure, which is subsequently subject to several improvements while using local search. All created candidates may be randomly added to a sorted list called RCL (Restricted Candidate List).

At each iteration of the randomized greedy algorithm, we randomly select from the restricted candidate list, a new candidate that we add to the solution, this random choice allows to modify the form of the constructed solutions, which are normally good, since the random choice is among a set of good.

This metaheuristic research method is based on two phases:

- Generation of a solution using the random and adapted greedy algorithm,
  - The greedy algorithm,
  - The adaptation function,
  - The selection with probability.
- Application of the local search on the resulting solution already generated in the first phase to see if there is still a possibility to be improved.

1.6.4.5 Hybrid algorithms

Hybrid algorithms are a combination of optimization techniques (exact methods, meta-heuristic algorithms) and are widely applied to solve complex problems (Blum, 2010), (Lozano, 2010), (Talbi, 2002). They can be classified as collaborative or integrative combinations (Raidl, 2005).

These combinations improve the solution quality while benefiting from the advantages of more than one algorithm at the same time; we can distinguish two types of combinations:

- The collaborative combination, which is, based on the sequential or synchronous exchange of information between some optimization techniques.
- The integrative combination of subordinating the embedded components of another technique, in this case, we have a dominant algorithm and other (s) dependent algorithm (s).

One of the most powerful combinations is those that combines Tabu Search and GRASP. (Laguna, 1991), were the first who study this maneuver. A few years later, this method became widely adopted and very useful in many fields, for example the rental problem (Juan A. Díaz, 2017).

Two strategies could be adopted to explain the principle of hybridization:

- ***Strategy 1:*** For example, we can approach the global optimum by a random method and then refine the result by successively applying a local method. This hybridization is called dumb hybridization because it applies one method after another. It is clear that the result will be better but at the expense of the calculation time; we will consume more time than if we had applied the two methods separately, but to benefit more from the hybridization, it is advantageous to apply a global method and find the right moment to switch with a local method.
- ***Strategy 2:*** We try to find a local minimum state and stay there only if it is the best of local minima (global). This strategy has been used in three global optimization methods:
  - The method of multiple initializations (multi start): We repeat several times the execution of a local optimization technique at different points of the space of values and we take the best result found.
  - Tunneling method: We launch the local search method to arrive at a local minimum; The Tunneling technique is used to eliminate all points in the space that have a value greater than the local minimum already found previously and then re-optimize to find another local minimum and so on.
  - The roller coaster method (Steepest Ascent Descent: SAD.) First, we look for a local minimum, then this time we look for a local maximum; this makes it possible to escape from the local minimum state. Then we redo a minimization followed by another maximization and so on. The smallest minimum found at the end, is candidate to be the global minimum.

1.7 Conclusion

In this chapter, we presented some types of existing logistics, and then we defined the supply chain with its different characteristics and its basic elements. Next, we presented the optimality problem and the different algorithms of exact resolution linear and binary linear models, but when the number of variables becomes too important, it is better to use a software that is capable to perform different iterations and to determine the optimal solution.

The different types of algorithms used for resolution are:

- Heuristic algorithms,
- Meta-heuristics algorithms,
- Hybrid algorithms

The objective of this part is to focus on the most used and the most known researching algorithms used to solve the existing optimization problems (exact and approximate methods), especially those in relation with the supply chain resolution. Finally, we discussed the principle of hybrid algorithms and their advantages.

Chapter 2

Thesis Positioning

Chapter II – Thesis Positioning

2.1 Introduction

For the past few years, many researchers studied the crisis management supply chain, more specifically; the human transportation problem (for evacuation purpose) had been widely studied and took the highest priority toward reaching optimality.

Many approaches appeared for resolving it within many fields whatever the catastrophe was naturally produced (flooding, earthquake...) or artificially made (nuclear explosion, war, terrorist attacks…). Whatever the disaster type was, all delivery deadlines should be respected. Note here that each disaster is unique and has its own characteristics (different resources and different priorities) and its own dependencies.

These varieties assure the complexity of the optimization problem, that's why the resolution requires a good and well-organized modeling based on supply chain strategy that insure the smooth routing of physical and informational flows.

2.2 Vehicle Routing Problem (V.R.P)

(Ramser, 1959) Proposes the Vehicle Routing Problem (V.R.P) to solve the problem of delivering gasoline to service stations. (V.R.P) is considered as a generalization of (TSP) (Traveling Salesman Problem) but the main difference between them is that the salesman must return to the starting location after some visited points (figure 5).

(V.R.P) is defined as follow:

We have "n" number of identical vehicles located in one central depot and their missions is to deliver distinct quantities of goods to m distant clients with fixed demand. The objective is to minimize the overall transportation cost by reducing, in first place, the overall transportation time (distance) in choosing the optimal route to be used by a group of vehicles to serve a group of customers and, in second place, by reducing the number of the required vehicles.

Let G be an oriented graph.

G = (N, A), where "N" is the set of nodes (customers and depot), and "A" the set of arcs that link between these nodes. Each arc (i, j) is associated with a cost $c_{i,j}$ and each costumer i ∈ N has a demand d_i.

(V.R.P) goal is to find a set of minimum cost vehicle routes that service every costumer such that:

- Each vehicle's route originates and terminates at the depot.

- Each customer is visited only once by exactly one vehicle and each vehicle services only one route.
- Each customer receives its demand.
- The capacity of each vehicle is respected.

However, when vehicles have maximum capacities, the total demand of each location should not exceed vehicle capacity and in this case, the (V.R.P) is called Capacitated Vehicle Routing Problems (C.V.R.P).

(V.R.P) is classified as a NP hard combinatorial optimization problem. While heuristic approaches do not guarantee optimality, they produce good results. Recently meta-heuristics methods have arisen and proven their efficiency as the most promising approaches that direct the research for the (V.R.P).

Practically, (V.R.P) is subject to further modifications in structure and to multiple additions in constraints, we can cite for example the (V.R.P) with time windows constraints (V.R.P.T.W) which specify the available delivery times at which a client is available to receive its demand (clients are geographically dispersed), also we can find the (V.R.P) with Pick-up and Delivery where each vehicle must visit the pick-up position before moving to the corresponding delivery location.

Many other alternatives of the (V.R.P) exist such as the multiple depot (V.R.P) where the company may user several depots to serve its customers and the open (V.R.P) where each vehicle is free to return to central depot after accomplishing its missions.

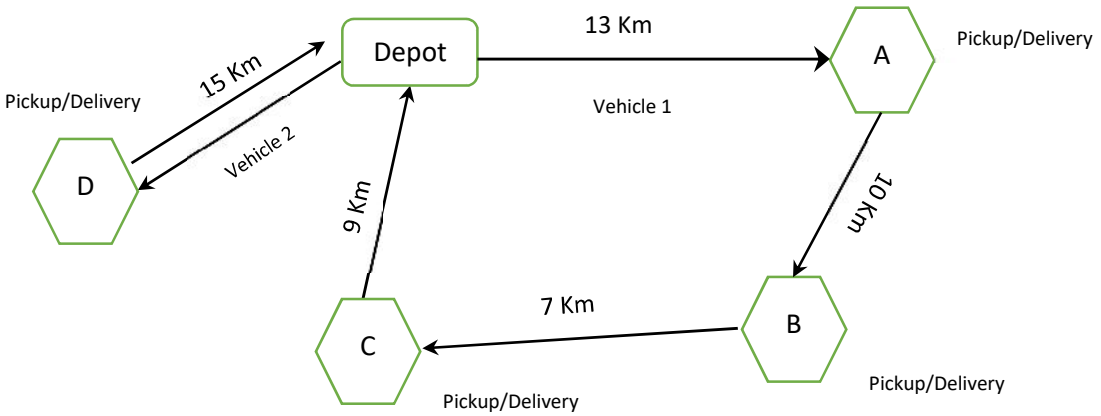


Figure 5. (V.R.P)

2.3 Vehicle scheduling problem (V.S.P)

Basely, V.S.P is used to design a set of routes that start from a central depot to serve multiple distributed customers at known places with a known demand; this design should take into consideration three restrictions:

- Customer satisfaction.
- Vehicle capacity.
- Route duration.

The main goal of (V.S.P) is to build a schedule of routes respecting these restrictions and minimizing the total number of routes in use and by consequences minimizing the total time spent. This objective (scheduling problems) is widely discussed by (Christofides N. M., 1989) (Christofides N. M., 1979) (Christofides N. M., Exact algorithms for the vehicle routing problem based on the spanning tree and shortest path relaxations, 1981) (Christofides N. M., State-space relaxation procedures for the computation of bounds to routing problems, 1981).

2.4 Vehicle routing and scheduling problems (V.R.S.P)

Routing and scheduling software must usually be customized to reflect the operating environment and the customer needs and characteristics. The Vehicle Routing and Scheduling Problem (VRSP) concerns the determination of routes and schedules for a fleet of vehicles to satisfy the demands of a set of customers (Abdelkader Sbihi, 2007).

VRSP are often presented as graphical networks

Objectives

a) minimize vehicles number
b) Minimize miles.
c) Minimize labor.
d) Satisfy service requirements.
e) Maximize orders.
f) Maximize volume delivered per mile.

2.5 Convergence with Supply Chain Management (S.C.M)

Supply Chain Management (SCM) is an essential element to operational efficiency; it requires the commitment of supply chain partners to work closely in coordinating order generation, order taking, and order fulfillment

In general, when a crisis is declared, too many constraints are presented and should be well respected in order to resolve the situation and return to the previous situation before crisis announcement (normal situation).

The supply chain is composed of two essential elements: Suppliers and clients.

In our study case, the situation is described as follow:

- We have multiple hospitals (suppliers) and their mission consist in providing local health care services for incoming patients.
- We have many wounded (clients) in need for these services and should be transported to hospitals.

In this particular case, we are not transporting goods from suppliers to clients, instead, we are moving clients (wounded) from their places toward hospitals using a capacitated vehicle (ambulances).

Therefore, we are dealing with a supply chain that starts with catering requests (transportation) announced by terminal zones (wounded) and addressed to supplier zones (hospitals) and ends with the reception of requested support (wounded).

During this operation, many steps are taking places. An agent must treat these demands, then he sends the requested support (wounded) via transport means (ambulances) to their final destinations (hospitals) while respecting predefined deadlines.

2.6 Related works

In this section, we present few works related to supply chain management and others dedicated to evacuation only.

Supply Chain Management (SCM) is an essential element to operational efficiency, it requires the commitment of supply chain partners to work closely to coordinate order generation, order taking, and order fulfillment.

Generally, natural and triggered disasters are unexpected and cause inevitable causalities. In many cases, people die due to deficiency in procedures such as evacuations plans. Evacuation Operations involve using available resources like ambulances, helicopters, etc. to transport causalities from disaster areas to safe places like hospitals, clinics, in order to improve life-savings.

In this regard, many researchers paid lot of attention to this topic.

(Christopher, From brand values to customer value, 1996), (M. Christopher, 1998) (Peck, 1-13) focused on the development of a managerial agenda for the identification and management of supply chain risk, with recommendations to improve the resilience of supply chains, their research has highlighted the risks to business continuity that lies in the wider supply chain. They defined the supply chain as a collected enterprises network that

participates, upstream and downstream, to different process and activities for creating products and services to end-consumer.

(R. R. Lummus, Strategic supply chain planning, 1998) Studied the relationship between information system and supply chain management (SCM) optimization. They define the supply chain as a network of entities in which the material flux passes through these entities, including providers, transporters, assembling sites, distribution centers, retailers and clients. To solve the supply chain problem, the authors offered a new strategy and some practical guidelines for successful supply chain management. They also described the competitive importance of linking a firm's supply chain strategy to its overall business strategy, but they did not take into consideration the fact of optimizing the transportation time spent in order to accomplish tasks.

(Tan, 2001) have developed a framework of supply chain management literature; they proposed various supply chain management strategies and the conditions conducive to supply chain management. Their contribution was illustrated by integrating the two bodies of literature (Purchasing and supply perspective of the industrial buyers and Transportation and logistics perspective of the merchants) in the unification of supply chain management into a commonly accepted terminology that includes all the value-creating activities along the value chain and finally they provided three distinct descriptions dominate prior literature:

- Supply chain management may be used as a handy synonym to describe the purchasing and supply activities of manufacturers.
- It may be used to describe the transportation and logistics functions of the merchants and retailers.
- It may be used to describe all the value-adding activities from the raw materials extractor to the end users and including recycling.

(Silva, 2006) worked on the distributed optimization problem of a logistic system and its providers. They improved the supply chain performance by introducing a new multi-agent approach for collaborative management of logistic and supply systems based on the ant colony optimization. The management methodology is defined as a set of distributed scheduling problems that exchange information during the optimization process. Each problem is solved by an ant colony agent that uses the pheromone matrix as the communication platform. Authors solved the whole problem by dividing it in several sub problems where each one is treated apart of the other. In our model, the problem is treated as one entity and all sub problems (tasks) are executed in coherence and harmony to reach the final goal.

(M. Chen, 2007) described a linear program to deal with multi-priority group evacuation in sudden onset disasters. The authors' goal is to minimize the total travel time to outside

the affected area, in their approach, they did not take into consideration the life expectancy of wounded and they have no plan to transport them to hospitals.

(Wei Yi, 2007) interested in studying the impact of a disaster and present for this reason a meta-heuristic approach grounded on the ant colony optimization to solve the logistics problem of relief activities which includes wounded evacuation to medical centers and dispatching commodities to distribution centers in crisis zones, their strategy depends on two phases of decision making. At first, they build stochastic vehicle paths, then at second, they solve the assignment problem between different types of vehicles flows and commodities, the authors suggest creating temporarily medical facilities in different places inside the affected areas in addition to the existing hospitals, also they plan to provide necessary goods to all existing facilities and transport all wounded toward them. The main objective was to minimize the delay between suppliers and distribution centers in the affected zones, to create temporarily medical facilities in addition to the permanent medical centers and to provide both of them with the necessary good and finally to transport the wounded to these medical centers. They are serving only severely and moderate injured people who are grouped in different places using different types of vehicles with different capacities to transport injured people.

(Zabinsky, 2010) proposed a stochastic optimization approach for the storage and distribution problem of medical supplies to be used for disaster management under a wide variety of possible disaster types and magnitudes. They developed a stochastic programming model to select the storage locations of medical supplies and required inventory levels for each type of medical supply by maintaining balance between the preparedness and the risk despite the uncertainties of disaster events. In their proposed model, they did not optimize the time spent by routing vehicles. They were only concerned in optimal routing plan of vehicles with their loading amounts of medical supplies in an MIP model.

(D.Eksioglu, 2010) used a mixed-integer linear program for evacuation plans of urban areas using public transit system; their objective is to minimize the total evacuation time and the number of casualties. The main difference between our model and their model is that all citizens are grouped in specific places (stations) and all transit vehicles had to collect citizens from the stations within its predefined lines to its closest shelter. In addition, in their model, they missed out the notion of priorities between stations and there is no plan to transport individual wounded to hospitals.

(Xin Ma, 2010) presented a wounded transportation model based on multi type vehicles to solve the wounded transfer problem in large-scale disasters. In their approach, each vehicle is assigned to a single disaster zone no matter how many wounded it contains and without

taking into consideration the time spent in a disaster zone which can cause that some vehicles which serve zones with little number of wounded will be free to serve again after some time, if opportunity is offered, meanwhile other vehicles which are serving zones with a huge number of wounded would stay busy for a longer time, as a result, as much as the number of wounded grows in zones as their evacuation time and their risk of death rise.

(João Coutinho-Rodrigues, 2012) presented a multi objective approach to generate urban evacuation plans where people are grouped inside a shelter and then they evacuate wounded to hospitals with different types of vehicles and there is no facilities capacities limitation. Using this strategy, they optimize the transport time but they risk the wounded life. Therefore, they are not reducing mortality rate because not all wounded can move to shelters which is not considered in their proposed model and which is the main objective in our proposed model.

(Noraimi Azlin Mohd Nordin, 2012) studied the fact of finding the shortest path for ambulance routing starting from ambulance station and ending at emergency site. They focused on finding the best roads network for the area under study (with the total shortest distance) using A* algorithm, in their model, there is no transportation plan given to ambulances from their emergency sites to medical treatment centers.

(Zhang, 2017) studied the fact of a disaster inside a high-rise building and defined a new evacuation device for emergency evacuation problem but, as before, they assumed that all people are grouped in the same place and they should all use the same route the same device without taking into consideration the human medical situation and the time spent to perform these actions.

(Keliang Chang, 2017) studied the impact of large disasters on roads safety and suggested for this reason a nonlinear site distribution for vehicle routing problem on the affected points.
In their model, they focused on transporting materials to uncertain demand points according to three objectives (1. Minimize vehicle total transportation cost, 2. Maximize the number of transported materials, 3. Maximize the poorest routes transportation capacities), but they did not rely on service priority for demand points.

(Yadollahnejad, 2017) proposed a model based on earthquake disaster and focusing on transferring wounded to clinics and hospitals using airlines and ground transportation. They introduced a multi-objective approach to provide transportation on available routes, but they group a certain number of wounded with different injuries in one vehicle in each transportation cycle and most wounded are located near to each other.

(Chawis Boonmee, 2017) worked on finding solutions for facility location problem in case where disasters are happening or already happened. They intended to find the best facility's location depending on many factors such as distribution centers, hospitals, etc. Unlike our model, they did not focus on the transported injured and they did not totally count on existent facilities (in our case: Hospitals) to which we intend to transport wounded, depending on many criteria.

(Huang, The impacts of sharing production information on supply chain dynamics: a review of the literature, 2003) have developed an integrated multi-objective optimization model for optimizing three objectives: minimizing human loss, optimizing delay cost and fairness, they combine resources allocation with emergency distribution in disaster response operations, in their study they aim to deliver urgent resources to in-site wounded in adequate quantity as fast as possible within a time barrier, they do not consider picking up wounded to hospitals or medical treatment centers. Other authors like (Byung Hoon Oh Kwangyeon Kim Han-Lim Choi, 2017) study the wounded evacuation problem after a natural disaster; they were interested in maximizing the number of evacuees by using helicopters with different capabilities. The main inconvenience of using helicopters is that they need a vast location to land, which may not be always available and sometime inapplicable. Some of the authors like (Wex, 2014), focused on emergency response in crisis management, they plan to minimize the sum of the total time spent in evacuating wounded outside disaster areas using different rescue units but don't have a plan to transport them to medical units. (Mori, 2015) have planned an evacuation model using multiple types of helicopters to maximize the number of transported wounded to secure areas, they took the time limit for survival as a hard constraint to be respected, but in their strategy, there is no plan to evacuate all victims to medical treatment centers.

First, in this section, we present some works related to supply chain management and others dedicated to evacuation only, after that, we justify our choice of adopted algorithms.

2.7 Proof of selected algorithms

The disaster area could be considered like a round field where all wounded are randomly distributed.
In reality, when an injury is critical, it means that the wounded have around 20 to 30 minutes to live, after that; he will be most probably dead.

We should note here that there is a wasted time calculated till first ambulance arrives to disaster zone and the associated doctor starts his mission (Wounded counting, wounded locations in order to provide environment parameters to end user agent who will enter

these data and launch the simulation software), so in fact the average remaining time for this case is decreased to less than 10 minutes.

Each tardiness second counts in losing an injured so the faster we provide results; the fewer wounded are dead.

When the execution time exceeds the 10 minutes' barrier, it means that the used method is useless and it should be aborted, that was the main reason that made us search for a very fast reliable algorithm, which can provide good solutions in a little time.

Due to the complexity of our proposed problem, which is NP-hard, we need to apply an intelligent fast algorithm to provide solutions to such problems.

As presented in the previous chapter number 1, many algorithms could be adopted to solve this kind of complex problems such as Genetic Algorithms (GA), Ant Colony Optimization (ACO), Simulated Annealing (SA), Tabu Search (TS), Greedy Randomized Adaptive Search Procedure (GRASP), and A Star algorithm etc.

Starting with (GA), one of the worst characteristic of this algorithm is its huge processing time that we do not have in our case of study; an additional cause is that its technique consists in crossing-over and mutating the selected initial solutions (parent solution) in order to create many neighborhood solutions (Childs). According to our study, these operations could not be applied due to several reasons. Mainly, the nature of our case requires a pre-monitoring of the situation before going into action and construct neighborhoods using crossover and mutation methods, these two methods are not applicable because our solution consists of several elements where their number depends on the studied situation, the main elements in use are hospitals, ambulances and wounded number. We cannot switch between hospitals randomly because hospitals subject to crossover/ mutation methods must have available sufficient serving capacities or the operation became infeasible and could not be done.

Continuing with (ACO) where ants should follow the highest deposited pheromone trail from the nest to food source. In our case, we may have multiple nests and multiple food sources. Each ambulance represents an ant, this ant should leave its nest once (ambulatory center) to the first food source (disaster zones or separated wounded) and then heads to the food source which is in our case (hospitals) which will be used as a new nest to start from and heads to another food source (another separated wounded or another disaster zone or maybe return to the same served disaster zone depending on the proposed situation). (ACO) strategy leads ants to find the best way to reach food source depending on the density of the deposited pheromone trail by exploring ants which, after finding it, will return to the nest and increases the deposited pheromone trail (subject to evaporation process). After that, the pheromone trail with the highest density means that its corresponding ant has

found the best path, the longest pheromone trail is subject to evaporation process when no ants follow it and increases its density. This strategy is inapplicable in our research study because we may have multiple separated distant wounded and in this particular case, each ambulance (ant) should serve only one wounded at a time to a hospital with available serving capacity and then move to another wounded to secure. In this case, the pheromone trail will always evaporate and there is no need to increase its density because the following ants has no wounded to secure.

Moving now to the third algorithm SA which could be more promising regarding our case, we are able to search the solution space with no fear of losing solution feasibility because all constraints are applicable in neighborhood construction phase, but the main disadvantage of this algorithm is its ability, according to our problem, to converge to a local optimum by not expanding its neighbor search in solution space. The temperature level, during search phase, decreases by level and in this case the exploration phase is being reduced slowly to the minimum and finally it stops and return the best solution found with no guarantee of optimality. Following this scenario, we can lose a promising neighbor due to temperature restriction and it converges easily to a local optimum. Another disadvantage of this algorithm is that its operations are extremely consuming time.

Continuing with Tabu Search algorithm, which is considered as one of the best-used technologies for solving difficult optimization problems that could not be solved with classical procedures in a much-reduced time. Glover was the first who developed a Tabu Search algorithm that has been effectively powerful and useful for solving combinatorial optimization problems. This technique relies on two phases: a. Adaptive memory; b. Responsive exploration; the first one record ancient search information depending on four important references: occurrence, recently visited, quality and influence. The second one is capable to make tactical selections to achieve effectiveness. The main role of using an adaptive memory is to record the pre-visited good candidates, this record is called Tabu list (forbidden list), its main purpose is to restrict choosing redundant candidates (marked as Tabu active) for some time, depending on the user pre-fixed Tabu list size, when the list size exceeds the user predefined limit, the first record is removed (aspiration method). The main importance of this adaptive memory is to avoid entering in a local optimum and to enhance the search by using intensification or diversification strategies, the intensification strategy allows the algorithm to exploit more the good found candidates while the diversification strategy lead to explore unvisited candidates inside the solution space.

Continuing with (GRASP) procedure that in the difference to A* algorithm it is capable to find quicker efficient solutions. After these previous experiments, we realize that if we need

to improve solution quality and respect processing time barrier, we are forced to adopt an efficient algorithm that prevent TS from getting stuck in a local optimum for a long time.

Ending with Hybrid algorithms, they are popular and well known for their efficiency compared to classic meta-heuristics. By using hybridization, we can avoid falling in some problems like the stagnation under local optima, also we can efficiency reduce the required processing time spent in calculating an optimal solution or an acceptable solution. Many hybridization strategies are developed in literature but we can identify two-man classes of hybridization. The first one is the hybridization by parallelization or by threading and it is a strategy based on the exploitation of concurrence and cooperation benefits when many meta-heuristic processes are launched in parallel. The second approach is the sequential hybridization and it is based on the alternate between two meta-heuristics or more and on a cooperation technique like the communication among some parameters. The hybrid algorithm, which we propose, is a sequential hybridization of (GRASP) algorithm and TS procedure. This approach is based on restarting the Tabu Search from different initial solutions generated by (GRASP) algorithm.

The best strategy found in literature was the sequential hybridization between (GRASP) and TS that allow TS, at each iteration, to improve the solution previously created and improved with (GRASP) procedure. At each iteration (GRASP) Explore neighbors in solution space and provide TS with the best-known solution to work on and improve its quality. This technique could lead to optimality in a much-reduced processing time.

2.8 Research motivation

Over the last few years many natural disasters, wars and terrorist attacks occurred and led to hundreds and thousands of wounded and killed people.

During Tōhoku Earthquake and Tsunami (2011)[iii], when an earthquake followed by a tsunami hit the east coast of Japan, 15,894 people were dead while 6,152 injured and 2,562 people missing. The Haiti Earthquake (2010)[iv], the strongest earthquake to hit the country since 1770, led to over 200,000 deaths, 2 million homeless, and 3 million people in need of emergency aid. Hurricane Katrina, (2005)[v], struck the Gulf Coast of the US, this was the sixth strongest and fifth most destructive hurricane to ever hit the US. It killed 1,833 people and left hundreds of wounded. In Iraq (2003)[vi], US troops started the invasion of Iraq in March 2003, in coalition with the UK and other nations. By 31 August 2010, when the last US combat troops left, 4,421 were killed. Almost 32,000 had been wounded in action. Similarly, an enormous loss was reported in Syria since the beginning of the conflict in 2011[vii], around 11.5% of the country's population have been killed or injured since the crisis erupted in March 2011, the report estimates that the number of wounded is put at 1.9

million. Life expectancy has dropped from 70 in 2010 to 55.4 in 2015.While in the Libyan armed conflict (2011)[viii], A total of 21,490 (0.5%) persons were killed and 19,700 (0.47%) were injured.

There is also another kind of disasters which are made by people like Terrorist attacks which hit multiple cities all over the world in different ways, such as car bomb, individual attacks etc. but all of them have the same consequences and common results revealed by a huge number of killed and wounded citizens. All these attacks are considered as terrorist acts and human disasters and treated as high-level crisis. For example: Paris attacks**:** On the 13th of November 2015[ix], six attacks in 33 minutes hit the capital of France, from France stadium to the capital center and caused the death for 129 citizens and hundreds of wounded spreads within this geographic area; most are seriously injured.

All these motivations encourage us to consider new technique that combines (C.V.R.P) with (V.S.P) in order to maximize life-saving and solve the integrated problem of ambulance scheduling and resources assignments.

2.9 Conclusion

In this chapter, we positioned our problem, then we list some related work found in the literature, after that, we described and compared several algorithms used to solve the Crisis management supply chain, and at the end of this chapter, we justified our choice of resolution algorithms and we explained why hybridization is more useful in our case of study.

Conclusion (First PART)

In this part, we investigate several algorithms used to attain optimal results. At first, we present some of the most popular exact methods cited in literature. Then at second, we present some of the related work found in literature. At third, we describe some of the most adopted metaheuristics algorithms that prove their efficiency in solving complex combinatorial problems. Moreover, at the end, we briefly present our case of study and validate our choice of algorithms.

In the next part, we present and discuss our case of study.

SECOND PART: MODELLING AND RESEARCH METHODOLOGY

The first chapter of this part (Chapter 3) focuses on solving the evacuation problem in its most difficult case where all wounded are separated and distant from each other and we plan to transport them, using ambulances, as fast as possible to distant hospitals sites. We propose a mathematical model followed by hybrid (GRASP-TS) algorithm that minimize the overall evacuation time and respect the different imposed constraints.

The second chapter (Chapter 4) describes another state of evacuation planning where wounded are grouped together in different disaster zones. Same constraints as the first approach are applicable. Our goal is to schedule ambulances missions and resources assignment in the best way in order to evacuate and secure these groups as fast as possible.

The third chapter (Chapter 5) represents a new case of study in a large-scale disaster where we can find separate or joined wounded in distant or close areas inside disaster zones. Our objectives are to create groups or divide (or not) existing groups into several sub groups at first, then, at second, solve the (IPAMSRA) counting on Image processing and other meta-heuristics methods.

Chapter III

A hybrid algorithm Tabu Search – GRASP for wounded evacuation in disaster response.

Chapter III – A hybrid algorithm Tabu Search – GRASP for wounded evacuation in disaster response.

3.1 Introduction

Natural and triggered-disasters have devastating and profound negative effects on human lives that require a speedy declaration of an emergency in order to minimize their severe consequences. Hence, a prompt disaster response, in addition to effective measures such as informed decision-making, organized evacuation plan, right hospital selection, proper rescue vehicles, efficient resources assignment and timely vehicle scheduling are critical actions needed to organize successful secured operations that could, if well prepared, save many injured bodies and lessen the human distress. To reach this ultimate goal, a complicated procedure should be in place and any failure can potentially increase the number of causalities, thus a complete alertness and full caution should be exercised.

In this Chapter, we treat the Integrated Problem of Ambulance Scheduling and Resource Assignment (IPASRA) in the case of a sudden disaster. The main resources to be assigned are the ambulances and the hospitals. While, the hospitals serving capacities might be considered or not according to the extent of disaster and particularly to the wounded bodies' total number. We formulate the (IPASRA) as a linear model, furthermore a novel hybrid algorithm based on Tabu Search (TS) and Greedy Randomized Adaptive Search Procedure (GRASP) is offered to tackle this complex problem. Simulation tests are also presented to prove the efficiency of our modelling and resolution approaches.

3.2 Literature Review

As mentioned before, some previous works are interested in solving the transport optimization problem inside a supply chain by reducing the transportation cost of goods according to predefined constraints and the others ones are interested in evacuating groups of wounded to existing hospitals or to in site temporary created clinics, but none of them consider picking up wounded one by one from different places or take the facilities capacities limitation in consideration.

Our approach deals with a complicated case of crisis management where we have multiple dispersed wounded and multiple distant hospitals. Our goal is to reduce the overall wounded's transport time from their locations to hospitals by ambulances in order to reduce mortality rate. Two realistic scenarios were treated; the first one considers using hospitals with limited serving capacity. This scenario is applicable when we are facing an emergency with a small number of wounded and where the total hospitals capacities can

cover this number. However, when the wounded number increases, the first scenario become useless and the second scenario without capacity constraint is being applied. We solve the (IPASRA) by developing a new algorithm using hybridization between two metaheuristic methods Tabu Search and (GRASP). The efficiency of the proposed hybrid algorithm in term of solution quality and processing time is shown in section 3.5.4.

3.3 Problem description and mathematical modelling

3.3.1 Problem description

As explained before, we deal with the Integrated Problem of Ambulance Scheduling and Resource Assignment where the resource assignment concerns the allocation of hospitals and ambulances to wounded. The situation is described as follow: we have a finite number of dispersed wounded, a fleet of ambulances located in several ambulatory centers and several hospitals in distant zones, we intend to transport the set of wounded to the hospital sites with the fleet of ambulances (Figure 6). Ambulance scheduling is defined by a sequence of tasks to be performed where each task has its own starting and ending locations. We consider two versions of the Integrated Problem of Ambulance Scheduling and Resource Assignment (IPASRA), the first one integrates a constraint of hospital capacity where each hospital treats a finite number of wounded and in the second one, that constraint is removed, in other terms with an infinite hospital capacity.

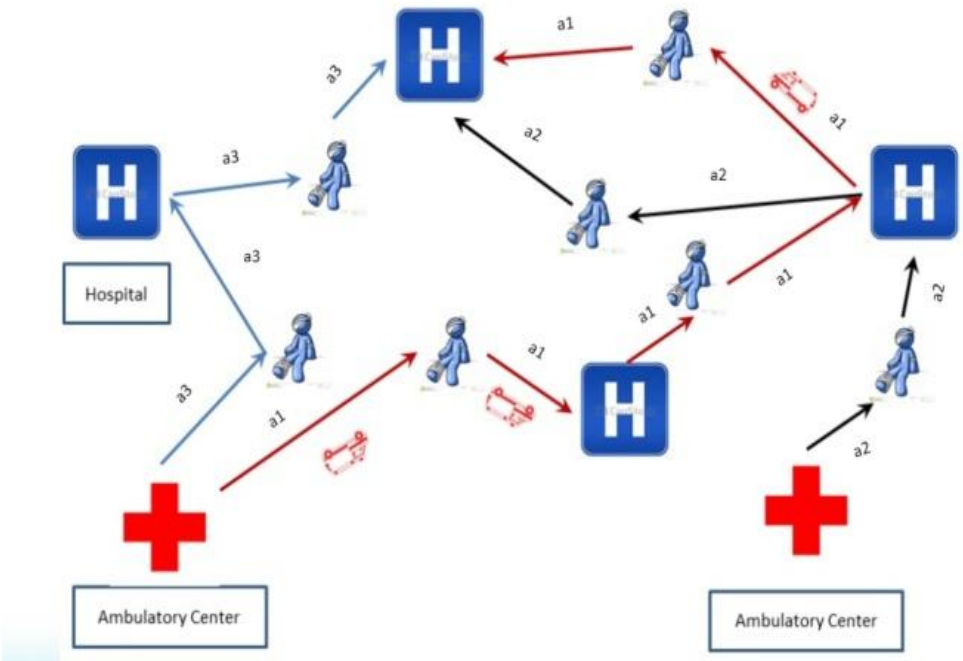


Figure 6. Distant wounded evacuation

3.3.2 Assumption

In this work, we assume that the following conditions are verified and respected.

- The total wounded number is known in advance.
- We use one type of vehicles, which is the ambulances, because they assure wounded safe transportation.
- We will assume that the total number of ambulances to be used for wounded transportation purpose is less than the total number of wounded, moreover we will limit the ambulance capacity to one wounded at a time (Standard capacity).
- We will set a finite capacity c_i for a hospital h_i.
- We will consider that the sum of all hospital's capacities is greater than the total wounded number.
- All wounded are distant from each other.
- We will not consider the wasted time spent by the operation of picking up and putting down (Load/Unload) wounded.
- Each hospital is capable to treat all kind of injuries.

The disaster area is considered like a round field where all wounded are distributed. In reality, when an injury is critical, it means that the wounded have around 20 to 30 minutes to live, after that; he will be most probably dead. Each tardiness second counts in losing an injured so the faster we provide results the fewer wounded are dead. Then we are interested in efficient solutions calculated with a processing time less than 10 min.

Due to the complexity of our proposed problem, which is NP-Complete as proved in section 3.4, we need to apply an efficient algorithm to provide solutions to such problems. Hence, a hybrid (GRASP-TS) algorithm is adopted to solve the problem; it will be described in the next section. In next part, we present (GRASP), Tabu Search and their hybridization and we discuss their adaption to our problem.

3.3.3 Mathematical Formulation

We present below the data used in our modelling.

- H is the set of hospital indexes and $|H|$ the number of hospitals.
- A is the set of ambulances indexes and $|A|$ the number of ambulances.
- P is the set of wounded indexes and $|P|$ the number of wounded.
- H_{fic} and P_{fic} are respectively the set of fictive wounded and the set of fictive hospitals. Each vehicle a is associated to a pair (p_a, h_a), where p_a is a fictive wounded and h_a a fictive hospital and where the distance between p_a and h_a is nil. These elements are used to insure the starting of each vehicle missions. The distance between h_a (or p_a) and each hospital h represents the distance between the depot of ambulance a and the hospital h. In others terms, p_a and h_a represent the depot of ambulance a.
- w_h theoretical capacity of hospital h, it is the maximal number of wounded which hospital h can serve at the beginning of the operation.
- t_0 operation's departure time.
- $T_{p,h}$: Matrix used to specify the total routing time spent in transporting the injured p to hospital h. This matrix is not symmetric in real cases.
- $T_{h,p}$: Matrix used to specify the total routing time spent in moving from the hospital h to the wounded p. We can consider here that h is the last visited hospital and p the next wounded to secure.
- G: a sufficiently large number.

In the following, we introduce the variables of the model.

C_{max}: the date of operation's ending called the makespan. This value is the objective to minimize in our modelling.

t_p: the time when the ambulance transporting wounded p arrive at hospital.

$X^a_{p,h,f}$: the only binary flow variable of our mathematical model is defined as follow

$X^a_{p,h,f}$ =
- 1 if the ambulance a transfer the wounded p to the hospital h and then move to secure the wounded f.
- 0 if else.

3.3.4 Modeling presentation

We present below the mathematical model of our problem. Note that as defined before, p_a is a fictive injured associated to ambulance "*a*" and h_aa fictive hospital associated to the same ambulance.

The Objective function C_M seeks to minimize the total time spent by ambulance routing.

Minimize C_M

Subject to the following constraints:

$$\sum_{p\in P} X^a_{p_a,h_a,p} = 1, \qquad \forall a \in A \qquad (1)$$

$$\sum_{a\in A}\sum_{p\in P}\sum_{f\in P, f\neq p} X^a_{p,h,f} = 1 \qquad \forall\, h \in H_f \qquad (2)$$

$$\sum_{a\in A}\sum_{h\in H}\sum_{f\in P, f\neq p} X^a_{p,h,f} = 1 \qquad \forall\, p \in P \qquad (3)$$

$$\sum_{a\in A}\sum_{h\in H}\sum_{f\in P, f\neq p} X^a_{f,h,p} = 1 \qquad \forall\, p \in P \qquad (4)$$

$$\sum_{h\in H}\sum_{f\in P, f\neq p} X^a_{p,h,f} = \sum_{l\in H}\sum_{g\in P, g\neq p} X^a_{g,l,p} \qquad \forall\, p \in P, \forall\, a \in A \qquad (5)$$

$$t_f \geq T_{p,h} + T_{h,f} + t_p + G\left(X^a_{p,h,f} - 1\right) \qquad \forall\, a \in A, \forall\, p \in P, \forall\, f \in P\backslash P_f \;, f \neq p \qquad (6)$$

$$C_M \geq t_p \qquad \forall\, p \in P\backslash P_f \qquad (7)$$

$$X^a_{p,h,f} \in \{0,1\}, t_p \in \mathbb{R}^+ \qquad \forall a \in A, \forall\, p, f \in P, \forall\, h \in H \qquad (8)$$

Constraint (1) ensures the starting of each ambulance mission by associating it to a fictive couple of injured and hospital. Note that these two fictive elements are situated in the same location as the associated ambulance at the beginning of its mission. The distance between the fictive injured and the fictive hospital is nil and the distance between the fictive hospital and each injured is equal to the distance between the initial location of the ambulance and the injured, which is initially known.

Constraint (2) ensures that each fictive hospital is to be visited only once. Constraint (3) ensures that every injured has only one direct successor and constraint (4) insures that he has only one predecessor. In other terms, constraints (3) and (4) ensure that each injured should be transported once. With constraint (5) we ensure that the successor and the predecessor of each injured are transported by the same ambulance. Constraint (6) insure the temporal succession by considering the binary variable $X^a_{p,h,f}$ and the traveling times $T_{p,h}$ and $T_{h,f}$, with this constraint, we evaluate the ending date of each securing mission denoted by t_p. Constraint (7) is

used to evaluate the makespan which is the time spent by the last mission ending. The makespan in denoted by C_M . Constraint (8) represents decision variables.

The evaluation of the makespan by constraint (6) and (7) represents a serious complication at simulation stage because of G that is a sufficiently large number. To improve the modelling, we replace constraint (6) and (7) respectively by constraint (9) and (10). We replace also t_p *by* R_p which represents a number associated to wounded p and used to avoid sub-cycles.

$$R_f \geq 1 + R_p + |P|\left(X^a_{p,h,f} - 1\right), \qquad \forall\, a \in A, \forall\, p \in P, \forall\, f \in P \backslash P_f \;\; , f \neq p \tag{9}$$

$$C_M \geq \sum_{p \in P} \sum_{h \in H} \sum_{f \in P \backslash P_f} X^a_{p,h,f}\,(T_{p,h} + T_{h,f}), \quad \forall\, a \in A \tag{10}$$

Constraint (9) is imposed to avoid sub-cycles and with constraint (10), the makespan is reduced by the total traveling time of each ambulance. By adopting constraints (9) and (10) instead of constraints (6) and (7), we improved the efficiency of the modelling in terms of simulation results. Note that the use of G in constraint (9) does not affects the quality of simulation results as in the case of constraint (6). Note that we integrated the ambulance scheduling and hospital affectation in the same variable $X^a_{p,h,f}$ to avoid the use of a big number in constraint (10).

Considering the presented constraints, the capacity of hospitals is not considered. According to crisis extent, the problem can be considered with or without capacity constraint. In fact, for a major disaster, when the number of injured is larger than the total capacity of hospitals, the capacity constraint should be ignored. However, in other situation, when the number of injured is less than the total capacity of all hospitals, it is more realistic to consider the capacity constraint in the modelling in order to propose an adequate solution. Capacity constraint is modelled by the following inequality.

$$\sum_{a \in A} \sum_{p \in P} \sum_{f \in P, f \neq p} X^a_{p,h,f} \leq w_h, \forall\, h \in H \backslash H_f \tag{11}$$

Our linear model is solved using CPLEX solver 12.6.

3.4 Prove of NP-Completeness

Consider a particular instance of (IPASRA) where the number of hospitals |H| is equal to the number of wounded |P| and where the distance between each injured and the nearest hospital is negligible compared to the distances between hospitals. For that instance, our problem is equivalent to the (V.R.P) where the vehicles are the ambulances

and the cities are the hospitals. The (V.R.P) is known to be NP-Hard, and then (IPASRA) is NP-Hard.

3.5 Resolution methods

At first, the proposed model is solved using CPLEX solver 12.6. As proven in section 3.4, our problem is Np-Complete and then exact solutions of large-scale instances are not practical. In this section, we develop our hybrid algorithm. For this part, we consider that the problem without capacity constraint is equivalent to the problem with infinite hospital capacity.

3.5.1 GRASP

We discuss now the adaptation of (GRASP) algorithm to solve our problem. As shown in next diagram, we start the algorithm by initializing some parameters like the routing time distances between wounded are essentially, ambulances and the best solution. Then we apply an iterative randomized procedure to construct a feasible solution. At each iteration of the randomized procedure, we treat the subsequent injured to secure and we select randomly one of the nearest ambulances and one of the nearest hospitals having sufficient capacity. Then we update the distances between the set of injured and the selected ambulance at this iteration. We update also the capacity of the selected hospital. When we finish the treatment of all wounded, we check if the constructed solution is better than all of the previous constructed solutions, if it was the case then we save it as the best solution and then we reinitialize the parameters of the randomized procedure and we restart it again. The number of randomized constructions is initially predefined, when it is reached, the procedure is stopped and the best solution found is returned.

3.5.2 Tabu Search

We present in the next paragraph the adaptation of Tabu Search approach to our problem.

We provide a detailed explanation for neighborhood construction after this general description. At first, let us define a negative iteration as an iteration that does not improve the best solution found. After that, we use the number of successive negative iterations as a diversification criterion, an intensification criterion and a stopping criterion. Before launching the Tabu Search procedure, an initial solution is generated and all of the required parameters are initialized such as the Tabu list, the current solution, the best solution and the number of

successive negative iterations. Next, we start the iterations of our algorithm. At each iteration, the neighborhood of the current solution is constructed.

The size of this neighborhood depends on the number of successive negative iterations at the current iteration. The best non-Tabu neighbor updates the current solution, if this neighbor is better than the best explored solution, then the best solution will be updated and the neighbor is appended to the Tabu list. If the maximal size of Tabu list is reached, the oldest element is removed. When the number of successive negative iterations is equal to a fixed integer that we denote by IMAX, the Tabu Search is stopped and the best solution is returned.

The neighborhood of each constructed solution (figure 7) is divided into three sub neighborhoods where each one is constructed according to a particular vision. For the first sub-neighborhood construction, the method is done by selecting a vehicle and replacing it with another one (figure 8). The wounded is being inserted in in the other vehicle-associated cycle. For the second sub-neighborhood construction (figure 9), we select a vehicle and we switch only the associated wounded with another wounded, in other term we switch service order between its associated wounded. For the construction of the third neighborhood (figure 10), we select a vehicle and we replace only the associated hospital with another available one and update their capacities.

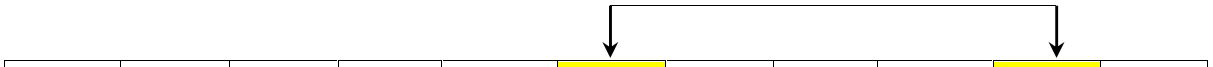
Neighborhood Construction:

Ambulance Number	Wounded Number	Hospital Number		Ambulance Number	Wounded Number	Hospital Number		Ambulance Number	Wounded Number	Hospital Number
0	2	1		1	1	0		1	0	2

Figure 7. Initial Solution

2	2	1		1	1	0		1	0	2

Figure 8. Neighbor 1



0	2	1		1	0	0		1	1	2

Figure 9. Neighbor 2

0	2	1		1	1	1		1	0	2

Figure 10. Neighbor 3

3.5.3 Hybrid Algorithm

With the following diagrams in figure 11 and in figure 12, we present respectively our (GRASP) procedure and hybrid algorithm with the integrated Tabu Search procedure.

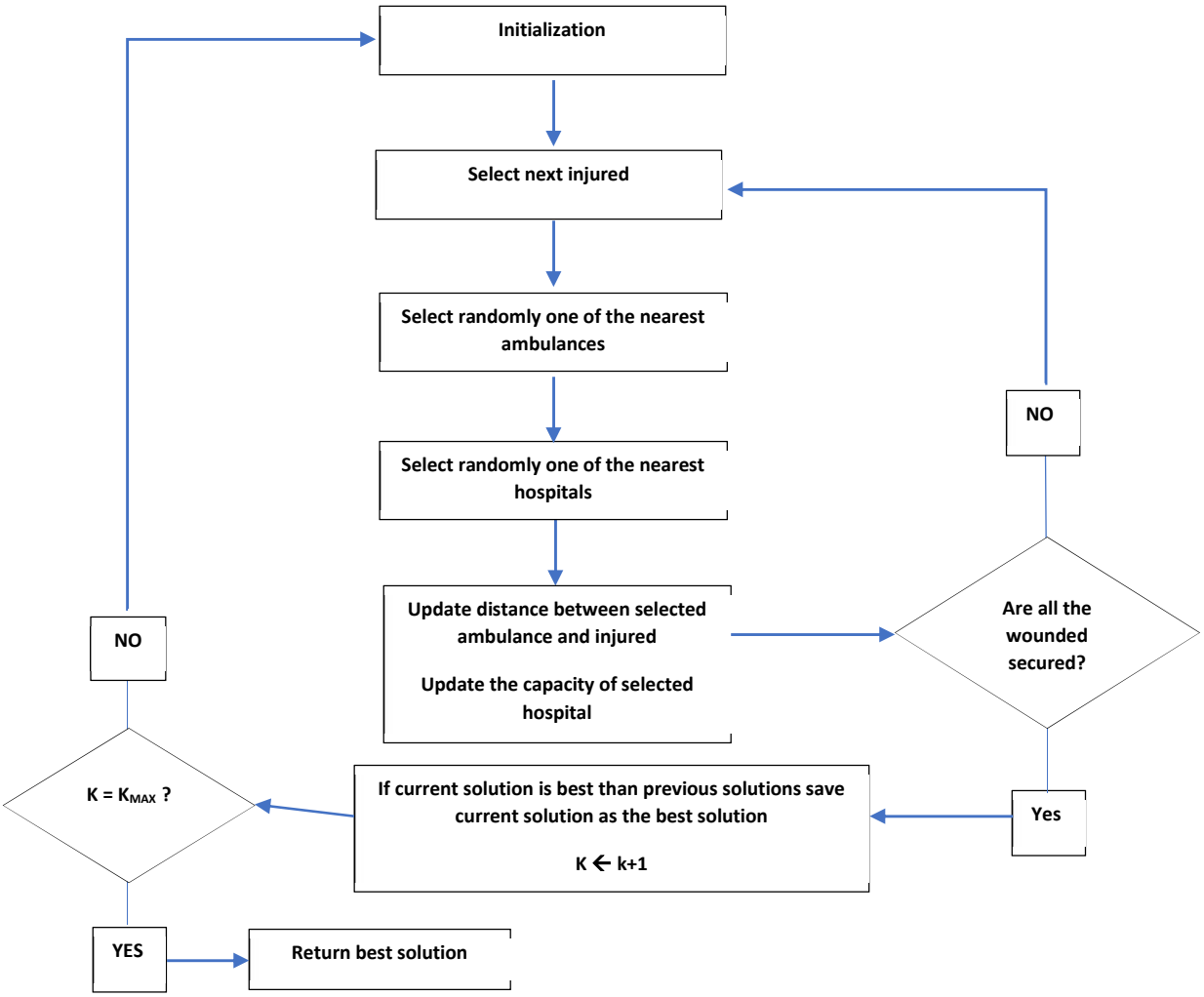


Figure 11. GRASP Algorithm.

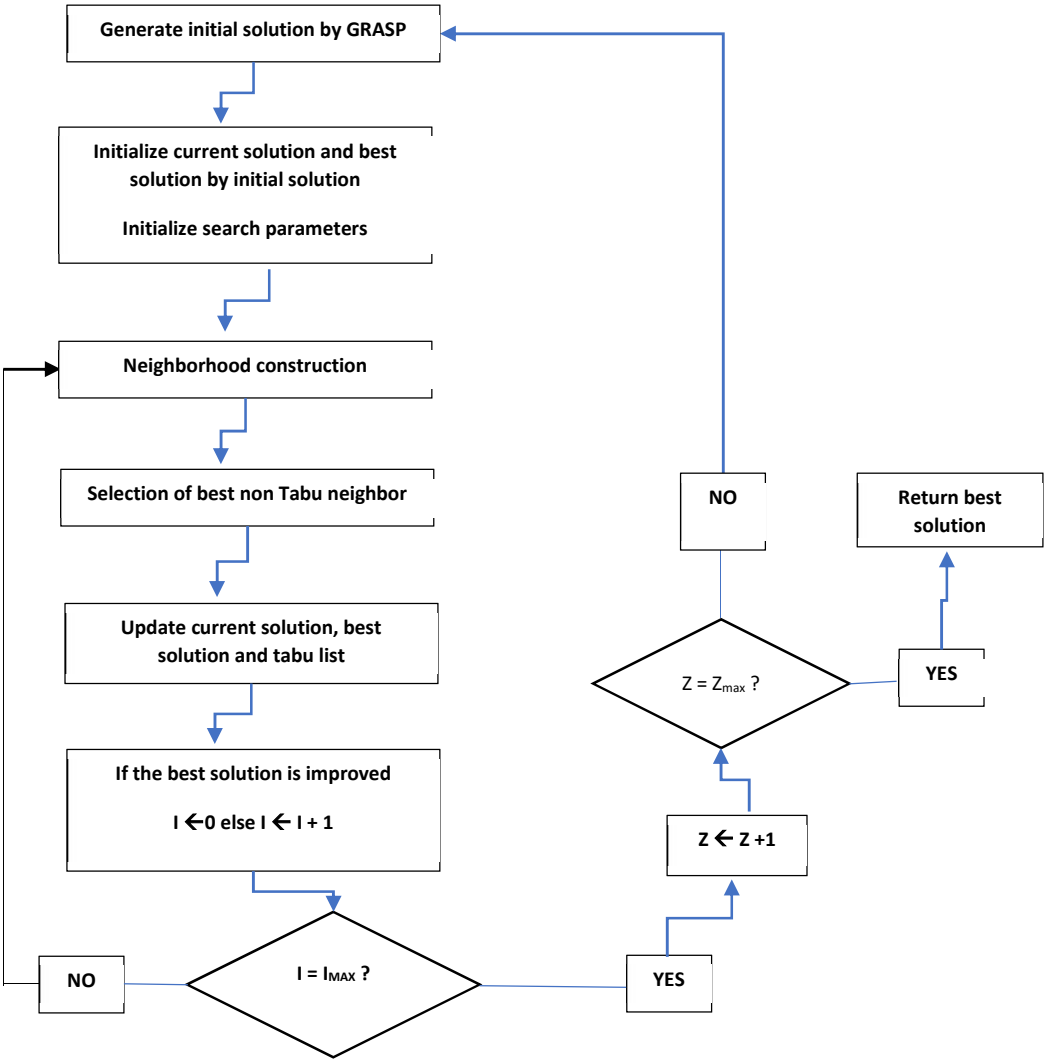


Figure 12. Hybrid GRASP-TS procedure.

3.5.4 *Experimental results and Discussions*

This section presents the obtained results for the instances of the (IPASRA). All Algorithms are coded using java, the computer used for simulation has a CPU Core I7 2.4 GHZ, 16 GB of RAM, 2GB dedicated VGA RAM and 500 GB SSD hard disk.

In Table 1, we present instances generated for simulation tests and for each one we give the number of wounded, the number of ambulances, the number of hospitals and the capacity of each hospital. Note that for every instance two other data are defined, the distances between hospitals and injured and the distances between ambulances and injured at the beginning of the secure operation. Instances are classified in three families F1, F2 and F3. F1 contains instances with

a number of injured equal to 10 or 15 wounded, F2 contains instances of 20, 30 or 50 injured and F3 is the family containing instances of more than 50 wounded.

The deviation between the CPLEX solution and the solution given by hybrid algorithm is calculated by:

$$\text{Gap} = \frac{V.H.A - V.C}{V.C}$$

V.H.A: Value given by Hybrid Algorithm

V.C: Value given by CPLEX

In table 2, we present the numerical results for capacitated (IPASRA). The simulation tests of non-capacitated (IPASRA) are presented in table 3.

Consider now the capacitated (IPASRA). For instances of family F1, the gap is between 0% and 0.94% and the processing time is less than 6 min. All the solutions given by CPLEX are optimal. From twelve instances, nine solutions calculated by hybrid algorithm are optimal and eight from them are given in less than 8 sec. For Family F2, Inst13, Inst14, Inst15 and Inst16 are solved optimally by CPLEX and the deviation between CPLEX solution and solution given by hybrid algorithm is less than 2%. These solutions are given in less than 23 sec. For the others instances of F2 the processing time is limited to 3600 secs and excepting Inst27 and Inst30, the CPLEX gap is less than 6 %. For Inst27 and Inst30 the CPLEX gaps are respectively equals to 9.59% and 20.88%. The deviation between these solutions and solutions given by hybrid algorithm is between -13% and 12%. For the instances of Family F3, the CPLEX gaps of inst31, inst32 and int33 are respectively 6.11 %, 9.73 % and 12.62 %. The deviations between CPLEX solutions and the solutions of hybrid algorithm are respectively 5.13 %, 3.62 % and -1.97 %. These deviations are reached after 10800 secs of running time. For instances from Inst34 to Inst40 no solution is given by CPLEX solver after 10800 secs of processing time. For Inst41, Inst42 and Inst43 no solution is returned because of the “out of memory”.

Note that deviation between CPLEX solutions and hybrid algorithm solutions is negative when the solution given by hybrid algorithm is better than the one given by CPLEX. Note also that solutions given by CPLEX are not optimal for some instances because of memory limit or processing time limitation.

Table 1. Problem characteristics.

Family	*Instance*	*/P/*	*/H/*	*/A/*
	Inst1	10	2	1
	Inst2	10	3	1
	Inst3	10	2	2
	Inst4	10	3	2
	Inst5	10	2	3
F1	*Inst6*	10	3	3
	Inst7	15	2	1
	Inst8	15	3	1
	Inst9	15	2	2
	Inst10	15	3	2
	Inst11	15	2	3
	Inst12	15	3	3
	Inst13	20	2	1
	Inst14	20	3	1
	Inst15	20	4	1
	Inst16	20	2	2
	Inst17	20	3	2
	Inst18	20	4	2
	Inst19	20	2	3
	Inst20	20	3	3
F2	Inst21	20	4	3
	Inst22	20	2	4
	Inst23	20	3	4
	Inst24	20	4	4
	Inst25	30	4	4
	Inst26	30	5	4
	Inst27	30	6	4
	Inst28	50	4	4
	Inst29	50	6	4
	Inst30	50	8	4
	Inst31	70	5	4
	Inst32	70	8	4
	Inst33	70	9	4
	Inst34	100	5	5
	Inst35	100	8	5
F3	Inst36	100	10	5
	Inst37	150	8	5
	Inst38	150	10	5
	Inst39	150	12	5
	Inst40	200	10	5
	Inst41	200	15	5
	Inst42	200	20	5

Table 2. Comparison of obtained results of applied methods with capacity constraint.

	Instance ID	*CPLEX Value (sec)*	*GRASP-TS Value (sec)*	*CPLEX CPU (sec)*	*TS-GRASP CPU (sec)*	*CPLEX GAP*	*TS-GRASP GAP*
F1	*Inst1*	11503	11503	< 1	< 1	0%	0%
	Inst2	3975	3975	< 1	2	0%	0%
	Inst3	2782	2782	2.8	48	0%	0%
	Inst4	3776	3776	0.81	6	0%	0%
	Inst5	4870	4870	1.2	5	0%	0%
	Inst6	2943	2943	7.64	317	0%	0%
	Inst7	4675	4675	0.52	88	0%	0%
	Inst8	13023	13023	0.31	32	0%	0%
	Inst9	2328	2328	172.29	234	0%	0%
	Inst10	8719	8721	123.07	48	0%	0.02%
	Inst11	3023	3024	53.13	104	0%	0.03%
	Inst12	1801	1818	355.7	313	0	0.94%
F2	*Inst13*	16816	16816	< 1	387	0%	0%
	Inst14	23487	23487	< 1	99	0%	0%
	Inst15	2892	2948	< 1	71	0%	1.94%
	Inst16	3497	3499	23	24	0%	0.06%
	Inst17	6733	6792	3600	195	1.26%	0.88%
	Inst18	8594	8753	3600	164	0.02%	1.85%
	Inst19	*10467*	*10463*	*3600*	*5*	*2.52%*	*-0.04%*
	Inst20	1762	1777	3600	54	0.51%	0.85%
	Inst21	2384	2413	3600	163	0.34%	1.22%
	Inst22	*8095*	*8085*	*3600*	*376*	*1.17%*	*-0.12%*
	Inst23	*6210*	*6165*	*3600*	*38*	*0.81%*	*-0.72%*
	Inst24	*6150*	*6113*	*3600*	*94*	*3.84%*	*-0.6%*
	Inst25	6450	6487	3600	329	1.01%	0.57%
	Inst26	*3572*	*3541*	*3600*	*111*	*5.49%*	*-0.87%*
	Inst27	3325	3444	3600	108	9.59%	3.58%
	Inst28	8949	9664	3600	387	1.74%	7.99%
	Inst29	4270	4762	3600	245	3.09%	11.52%
	Inst30	*6590*	*5779*	*3600*	*242*	*21.88%*	-12.31%
F3	*Inst31*	*8433*	*8866*	*10800*	*61*	*6.11%*	5.13%
	Inst32	*7490*	*7761*	*10800*	*188*	*9.73%*	3.62%
	Inst33	7926	7770	10800	200	12.62%	-1.97%
	Inst34	N.S	15552	10800	40		
	Inst35	N.S	4924	10800	279		
	Inst36	N.S	9498	10800	570		
	Inst37	N.S	15438	10800	734		
	Inst38	N.S	17249	10800	305		
	Inst39	N.S	10159	10800	634		
	Inst40	N.S - O.M	20197	9673	866		
	Inst41	N.S - O.M	16017	< 1	625		
	Inst42	N.S - O.M	7874	< 1	820		

Consider now the non-capacitated (IPASRA) given in table 3. Starting with instances of family F1, the CPLEX gap is between 0.8% and 1.78 %, 10 of 12 instances were optimal, 8 of them were given in less than 3 minutes and the remaining 2 were given in less than 44 minutes, Inst7 and Inst11 didn't attain optimality after a fixed processing time of 1 hour with a gap equals to 1.78 % and 0.17 % respectively. With the hybrid algorithm, from a total of 12 instances, 9 were optimal in less than 3 minutes. For both Inst7 and 1 Inst 1 the hybrid algorithm produces exactly the same value as CPLEX in less than 5 sec. Finally, for Inst11, a deviation of 0.14% was given by hybrid algorithm with a processing time equal to 38 secs. Continuing with F2 family, Inst13, Inst14, Inst17, Inst20, Inst21, Inst26 and Inst27 were solved optimally by CPLEX in less than 345 secs and the deviation between CPLEX solutions and the solutions given by hybrid algorithm is equal to 0% for all instances except for Inst27 where the deviation is equal to 3.13 %, all these solutions are given in less than 6 minutes. For instances Inst15, Inst16, Inst18, Inst19, Inst22, Inst23, Inst24, Inst25, Inst28, Inst29 and Inst30 the CPLEX gap is between 0.07 % and 10.91 %. The deviation between these solutions and solutions given by hybrid algorithm is between -3.75 % and 4 %. Ending with instances of family F3, the CPLEX gaps of Inst31 and Inst32 are respectively 0.06 % and 1.91 %. For Inst33 the CPLEX gap is equal to 64.68 %, these solutions were attained after 10800 secs of CPU Consuming time. The deviations between these solutions and the solutions produced by the hybrid algorithm are equal to 2.86 %, 0.55 % and -60.81 % respectively. These solutions are reached in less than 6 minutes of processing time. For instances Inst34, Inst35, Inst36, Inst37, Inst 38, Inst39 and Inst40 there is no solution given by CPLEX after 10800 secs and finally for Inst41 no solution is returned due to "out of memory" but with a processing time equal to 1499 sec, meanwhile for inst42 the "no solution out of memory" is returned directly.

We Should note here that when the deviation between CPLEX solutions and Hybrid solutions is negative, it means that the produced solutions by hybrid algorithm is better than the ones produced by CPLEX which are not optimal due to processing time limitation or due to memory limitation.

Table 3. Comparison of obtained results of applied methods without capacity constraint.

	Instance ID	*C-VAL (sec)*	*H-VAL (sec)*	*C-CPU (sec)*	*H-CPU (sec)*	*C-GAP*	*H-GAP*
F1	*Inst1*	11275	11275	< 1	< 1	0%	0%
	Inst2	948	948	< 1	< 1	0%	0%
	Inst3	2259	2259	< 1	< 1	0%	0%
	Inst4	3595	3595	< 1	< 1	0%	0%
	Inst5	4839	4839	2595	< 1	0%	0%
	Inst6	1878	1878	3	< 1	0%	0%
	Inst7	3487	3487	3600	< 1	1.78%	0%
	Inst8	9856	9856	< 1	< 1	0%	0%
	Inst9	2325	2325	69	5	0%	0%
	Inst10	8602	8614	1018	38	0%	0.14%
	Inst11	2950	2950	3600	5	0.17%	0%
	Inst12	1799	1799	151	170	0%	0%
F2	*Inst13*	11338	11338	2.64	< 1	0%	0%
	Inst14	9030	9030	1.39	< 1	0%	0%
	Inst15	2826	2844	3600	145	0.07%	0.64%
	Inst16	2398	2398	3600	9	0.54%	0%
	Inst17	1131	1131	41	<1	0%	0%
	Inst18	7971	7971	3600	51	0.84%	0%
	Inst19	10463	10463	3600	12	4.73%	0%
	Inst20	852	852	66	< 1	0%	0%
	Inst21	1213	1213	299	345	0%	0%
	Inst22	7123	7123	3600	4	3.47%	0%
	Inst23	*6122*	*6101*	*3600*	*63*	*1.08%*	-0.34%
	Inst24	4077	4099	3600	237	1.03%	0.54%
	Inst25	*5158*	*5133*	*3600*	*281*	*3.76%*	-0.48%
	Inst26	732	732	43	45	0%	0%
	Inst27	992	1023	96	13	0%	3.13%
	Inst28	6045	6126	3600	287	0.88%	1.34%
	Inst29	*3904*	*3748*	*3600*	*241*	*10.91%*	-4%
	Inst30	*3416*	*3288*	*3600*	*317*	*5.50%*	-3.75%
F3	*Inst31*	1732	1832	10800	314	0.06%	2.86%
	Inst32	2198	2210	10800	113	1.91%	0.55%
	Inst33	9465	3709	10800	121	64.68%	-
	Inst34	N.S	9564	10800	317		
	Inst35	N.S	1610	10800	25		
	Inst36	N.S	7480	10800	216		
	Inst37	N.S	2110	10800	274		
	Inst38	N.S	17249	10800	305		
	Inst39	N.S	3564	10800	709		
	Inst40	N.S	2629	10800	414		
	Inst41	N.S – O.M	12912	1499	511		
	Inst42	N.S – O.M	1926	< 1	619		

In figures 13 and 14, we can clearly recognize the huge difference in CPU consuming Time between CPLEX and GRASP-TS for both scenarios.

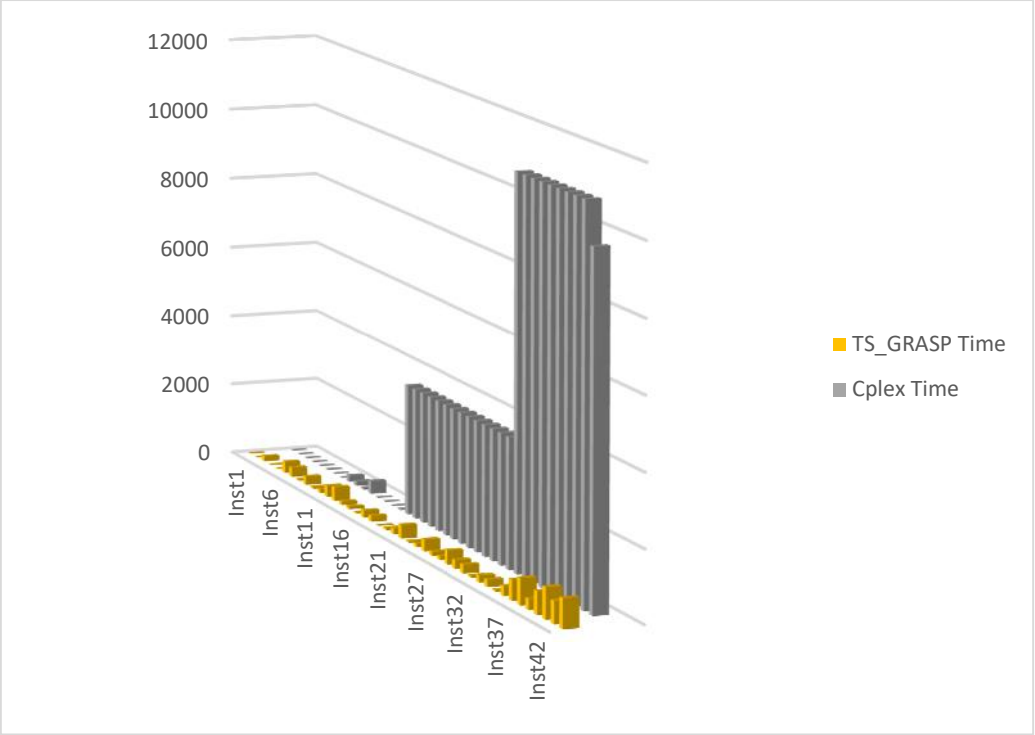


Figure 13. CPLEX vs Hybrid Algorithm – comparison of processing time / with capacity constraint.

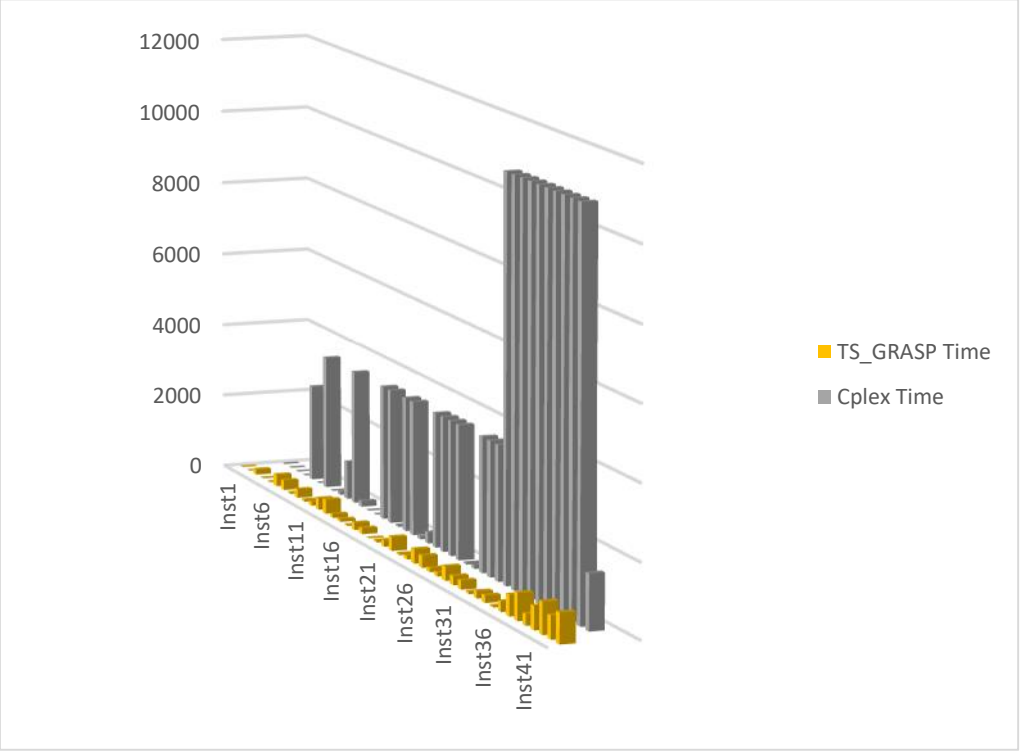


Figure 14.CPLEX vs Hybrid Algorithm – comparison of processing time / without capacity constraint.

In figures 15 and 16 we can clearly distinguish, in both scenarios, the values convergence between CPLEX and (GRASP-TS), also we can visually notice the CPLEX limitations.

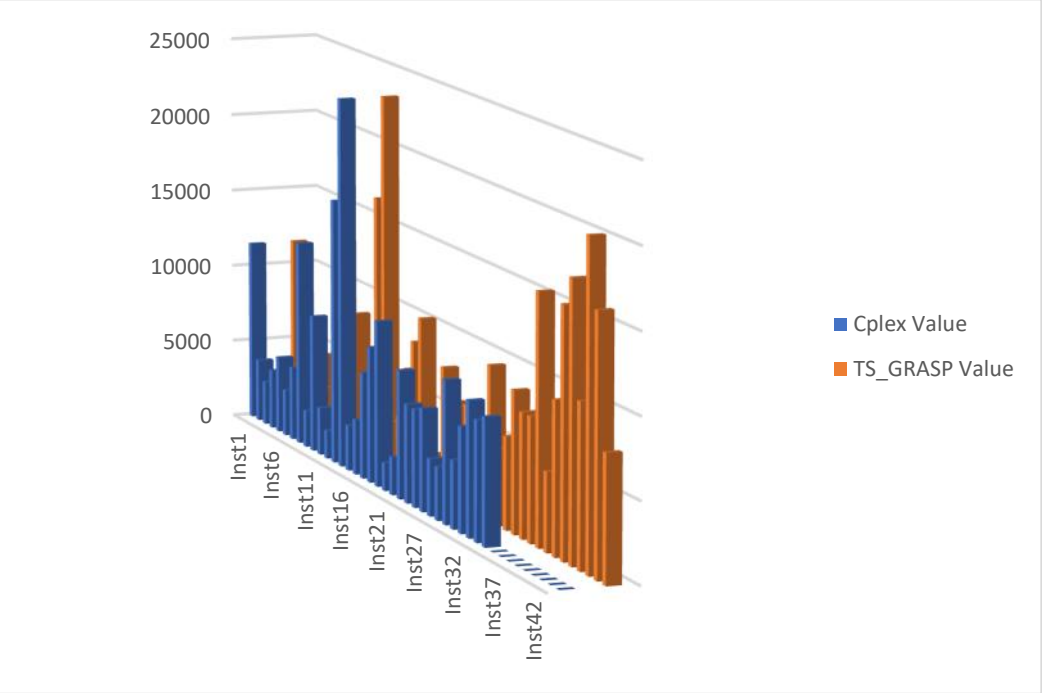


Figure 15. CPLEX vs Hybrid Algorithm – comparison of solution value / with capacity constraint.

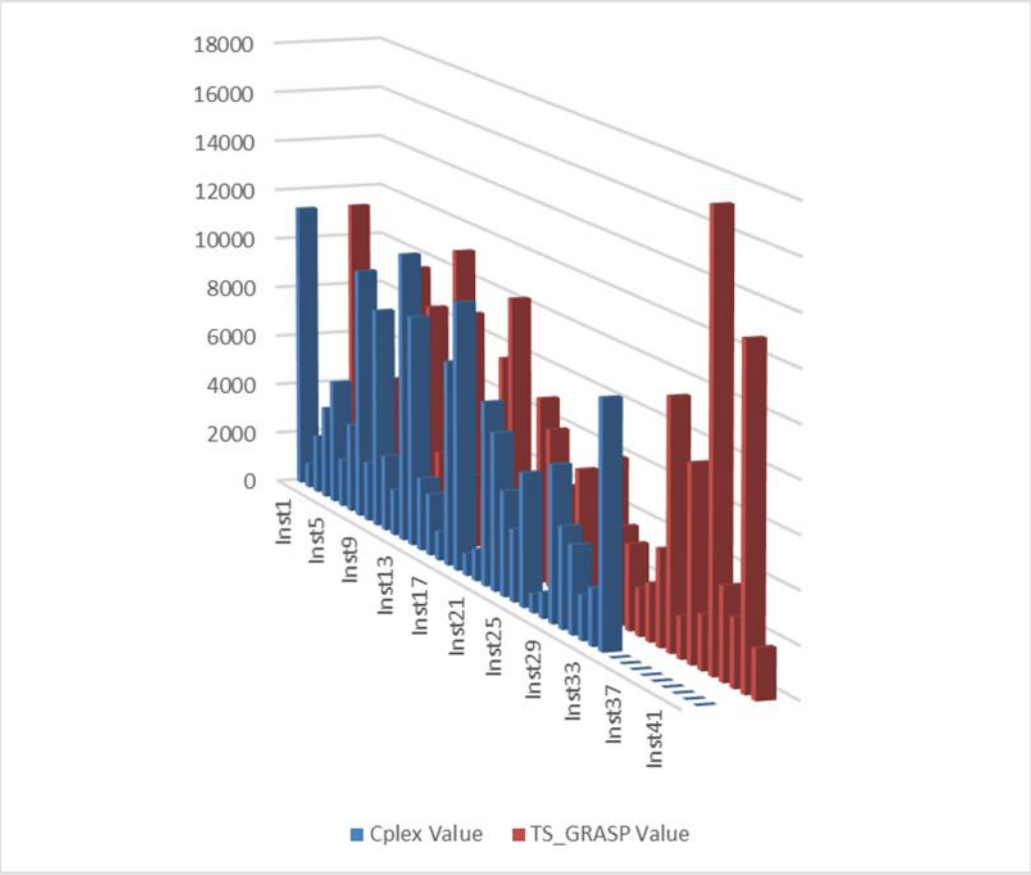


Figure 16. CPLEX vs Hybrid Algorithm – comparison of solution value / without capacity constraint.

3.6 Failure tests

Note here, before attaining the previous results, many failure cases have occurred. At first, we tried to apply TS algorithm alone and we implement it after providing it an initial solution using A Star method. Results were very promising in small and mid- size instances but when we move to big-size instances the TS deviation increases quickly as shown in the following tables.

In these tables, first column shows the instance size, second and third ones show the obtained objective value in CPLEX and TS respectively, the fourth and fifth columns show the elapsed runtime in CPLEX and TS respectively and finally we show the gap for CPLEX and TS in columns 6 and 7 respectively.

All instances are defined as follow: Instance name Wm_An_Hp means that the number of wounded W is equal to m, the number of used ambulances A is equal to n and finally the total number of hospitals H is equal to p.

To test the performance of TS we implement different instances with large problem sizes, Results of large instances with same number of hospitals are treated.

In all tables, we consider the following definitions:

C.V: CPLEX Value
TS.V: Tabu Search Value
C.CPU: CPLEX CPU Consuming time.
TS.CPU: Tabu Search CPU Consuming Time.
C Gap: CPLEX Gap.
TS Gap: Tabu Search Gap.

The Deviation in Tabu Search is given using this formula

$$\text{Gap}= \frac{T.V - C.V}{C.V}$$

In Table 4 (part I, II and II), we present the numerical results for capacitated TS. From 95 instances TS provides 30 instances with optimal values in a very reduced time less than 390 sec, for the remaining instances TS provides efficient solutions with gap value between 0.12% and 5.4%, CPU processing time was always less than 390 sec. which prove the efficiency of TS.

Table 4.Small size instances. Part I

Instance ID	C.V	TS. V	C.CPU (sec)	TS. CPU (sec)	C Gap	TS Gap
W10_A1_H2	11503	11503	0.03	0	0%	0%
W10_A1_H3	3975	3975	0.31	2	0%	0%
W10_A2_H2	2259	2259	1.44	4	0%	0%
W10_A2_H3	3595	3595	0.27	0	0%	0%
W10_A3_H2	4839	4839	0.67	1	0%	0%
W10_A3_H3	1878	1878	1.44	0	0%	0%
W11_A1_H2	8904	8904	0.05	0	0%	0%
W11_A1_H3	4527	4527	0.05	5	0%	0%
W11_A2_H2	8849	8849	0.8	2	0%	0%
W11_A2_H3	1454	1454	176.52	4	0%	0%
W11_A3_H2	4741	4741	37.3	5	0%	0%
W11_A3_H3	2037	2037	9.56	5	0%	0%
W12_A1_H2	11402	11402	0.01	0	0%	0%
W12_A1_H3	2613	2613	0.05	5	0%	0%
W12_A2_H2	7824	7839	64.58	147	0%	0.19%
W12_A2_H3	3169	3169	19.16	5	0%	0%
W12_A3_H2	720	720	0.45	7	0%	0%
W12_A3_H3	948	950	2.91	21	0%	0.21%
W13_A1_H2	16584	16584	0.06	0	0%	0%
W13_A1_H3	4304	4304	64.03	28	0%	0%
W13_A2_H2	6093	6129	103	72	0%	0.59%
W13_A2_H3	838	838	336.47	0	0%	0%
W13_A3_H2	4855	4855	4.3	5	0%	0%
W13_A3_H3	2352	2352	48.14	60	0%	0%
W14_A1_H2	12156	12156	0.02	116	0%	0%
W14_A1_H3	8596	8596	0.09	2	0%	0%
W14_A2_H2	1220	1220	0.28	3	0%	0%
W14_A2_H3	2380	2380	0.55	1	0%	0%
W14_A3_H2	6436	6436	11.69	120	0%	0%
W14_A3_H3	4922	4922	280.98	20	0%	0%
W15_A1_H2	3487	3487	16702	17	0%	0%
W15_A1_H3	9967	9967	0.13	2	0%	0%
W15_A2_H2	2325	2325	84.53	154	0%	0%
W15_A2_H3	8602	8614	8743	147	0.13%	0.14%
W15_A3_H2	2950	2950	9871.48	3	3.65%	0%
W15_A3_H3	1799	1805	130.71	0	0%	0.33%
W16_A1_H2	21876	21876	0.06	117	0%	0%
W16_A1_H3	10676	10676	0.09	23	0%	0%
W16_A1_H4	11981	11993	0.39	40	0%	0.10%
W16_A2_H2	3749	3792	329.86	56	0%	1.15%
W16_A2_H3	2034	2034	65.29	148	0%	0%
W16_A2_H4	4996	5090	7.85	197	0%	1.88%

Table 4.
Small size instances. Part II

Instance ID	C.V	TS. V	C. CPU (sec)	TS. CPU (sec)	C Gap	TS Gap
W16_A3_H2	9100	9100	23790.15	107	1.84%	0%
W16_A3_H3	1889	1908	662.22	51	0%	1.01%
W16_A3_H4	1952	1978	4236.16	58	0%	1.33%
W16_A4_H2	5465	5465	5.79 sec	2	0%	0%
W16_A4_H3	2257	2257	21809.66	198	0.53%	0%
W16_A4_H4	743	744	704.67	60	0%	0.13%
W17_A1_H2	6104	6104	0.03	4	0%	0.00%
W17_A1_H3	15722	15765	0.14	6	0%	0.27%
W17_A1_H4	16827	16827	0.34	2	0%	0%
W17_A2_H2	10687	10702	1007.08	100	0%	0.14%
W17_A2_H3	934	982	31358.27	329	0%	5.14%
W17_A2_H4	4993	5039	59.41	34	0%	0.92%
W17_A3_H2	8487	8484	71245.34	5	5.21%	-0.04%
W17_A3_H3	5897	5900	3457.76	286	0.02%	0.05%
W17_A3_H4	3142	3138	3603	3	0.11%	-0.13%
W17_A4_H2	2552	2613	528.53	262	0%	2.39%
W17_A4_H3	1335	1336	234.66	4	0%	0.07%
W17_A4_H4	1709	1709	3600.36	275	1.22%	0%
W18_A1_H2	19570	19570	0.09	53	0%	0%
W18_A1_H3	12058	12065	1679.53	37	0%	0.06%
W18_A1_H4	2124	2137	0.31	300	0%	0.61%
W18_A2_H2	4947	4947	2.8	8	0%	0%
W18_A2_H3	6161	6161	3600.02	333	0.05%	0%
W18_A2_H4	10973	11116	3600.01	352	0.04%	1.30%
W18_A3_H2	9428	9454	3600.03	161	0.14%	0.28%
W18_A3_H4	3018	3138	3601.3	51	0.73%	3.98%
W18_A4_H2	8172	8170	3600.31	53	4.58%	-0.02%
W18_A4_H3	2140	2140	2561.22	0	0%	0%
W18_A4_H4	3078	3131	3600.55	302	1.12%	1.72%
W19_A1_H2	18356	18356	3600.22	27	0.02%	0%
W19_A1_H3	8707	8709	0.27	70	0%	0.02%
W19_A1_H4	16113.99	16179	3600.43	97	0.07%	0.40%
W19_A2_H2	13040	13051	3600.25	88	0.04%	0.08%
W19_A2_H3	12478	12546	3600.02	33	0.03%	0.54%
W19_A2_H4	14044	14044	3600.55	130	1.43%	0%
W19_A3_H2	5646	5675	3600.44	242	0.19%	0.51%
W19_A3_H3	6092	6164	3601.67	140	0.59%	1.18%
W19_A3_H4	7611	7620	3604.55	65	1.18%	0.12%
W19_A4_H2	4201	4201	3600.23	10	4.40%	0%
W19_A4_H3	3076	3039	3601.42	10	2.43%	-1.20%
W19_A4_H4	2760	2797	3600.25	250	1.29%	1.34%
W20_A1_H2	16816	16816	0.08	387	0%	0%
W20_A1_H3	23487	23487	0.14	99	0%	0%
W20_A1_H4	2892	2948	0.38	71	0%	1.94%
W20_A2_H2	3497	3499	23.25	24	0%	0.06%
W20_A2_H3	6733	6792	3600.56	195	1.26%	0.88%
W20_A2_H4	8594	8753	3600.08	164	0.02%	1.85%

Table 4.
Small size instances. Part III

Instance ID	C.V	TS. V	C. CPU (sec)	TS. CPU (sec)	C Gap	TS Gap
W20_A3_H2	10467	10463	3603.59	5	2.52%	-0.04%
W20_A3_H3	1762	1777	3600.78	54	0.51%	0.85%
W20_A3_H4	2384	2413	3600.69	163	0.34%	1.22%
W20_A4_H2	8095	8085	3600.95	376	1.17%	-0.12%
W20_A4_H3	6210	6165	3605.2	38	0.81%	-0.72%
W20_A4_H4	6150	6113	3604.73	94	3.84%	-0.60%

Moving now to table 5 with medium instances size. From a total of 8 instances TS provides inefficient solutions for 6 of them and the deviation value was varying between 4% and 18% even with more CPU consumption time.

These instances represent a failure cases for TS regarding our problem

Table 5. Medium-Size Instances

Instance ID	C.V	TS. V	C. CPU (sec)	TS. CPU (sec)	C Gap	TS Gap
W25_A4_H4	4344	4733	122073.09	390	0.83%	9%
W30_A4_H4	6433	6487	31641	329	0.75%	1%
W30_A5_H4	3572	3541	3600.06	111	5.49%	-1%
W30_A6_H4	3325	3444	3600.09	108	9.59%	4%
W50_A4_H4	8949	9664	3605.47	387	1.74%	8%
W50_A6_H4	4270	4762	3600.36	245	3.09%	12%
W50_A8_H4	6590	7779	3600.38	242	21.88%	18%
W70_A5_H4	8867	9866	3600.28	61	10.70%	11.27%
W70_A8_H4	N. S	2212	3600.67	271		
W70_A9_H4	N. S	3777	3600.25	57		
W100_A5_H5	N. S	-	3600	-		
W100_A8_H5	N. S	-	3600	-		
W100_A10_H5	N. S	-	3600	-		
W150_A8_H5	N. S	-	3602.69	-		
W150_A10_H5	N. S	-	3600	-		
W150_A12_H5	N. S	-	3600	-		
W200_A10_H5	N. S	-	9672.59	-		
W200_A15_H5	O.M		<1			
W200_A20_H5	O.M		<1			

These results are graphically shown in figures 17, 18 and 19.

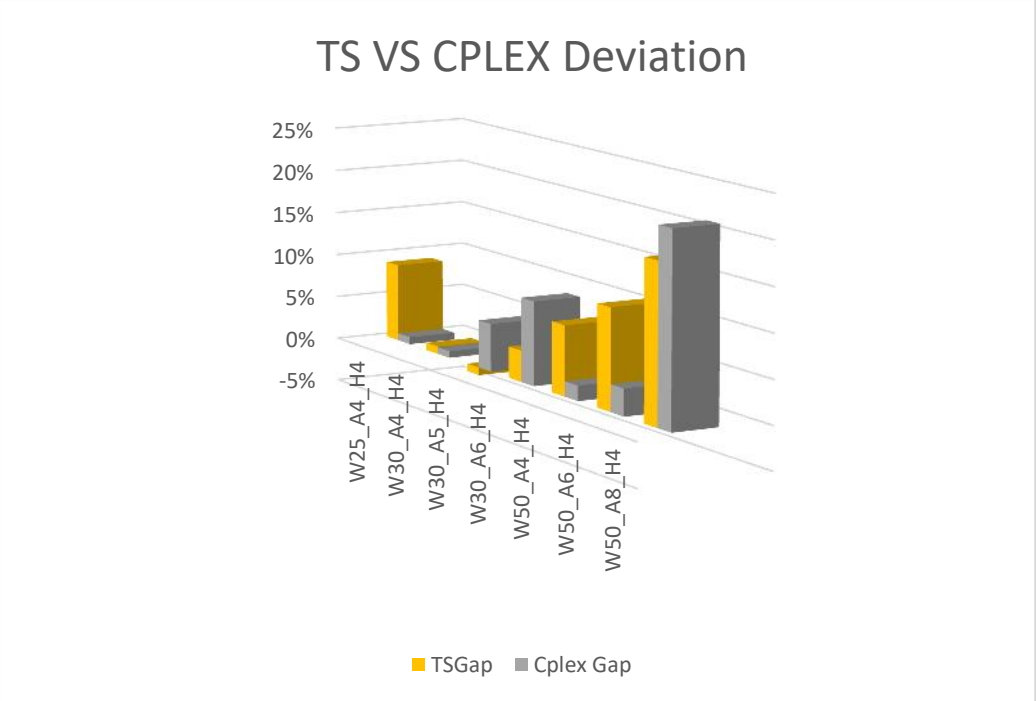


Figure 17. TS VS CPLEX deviation.

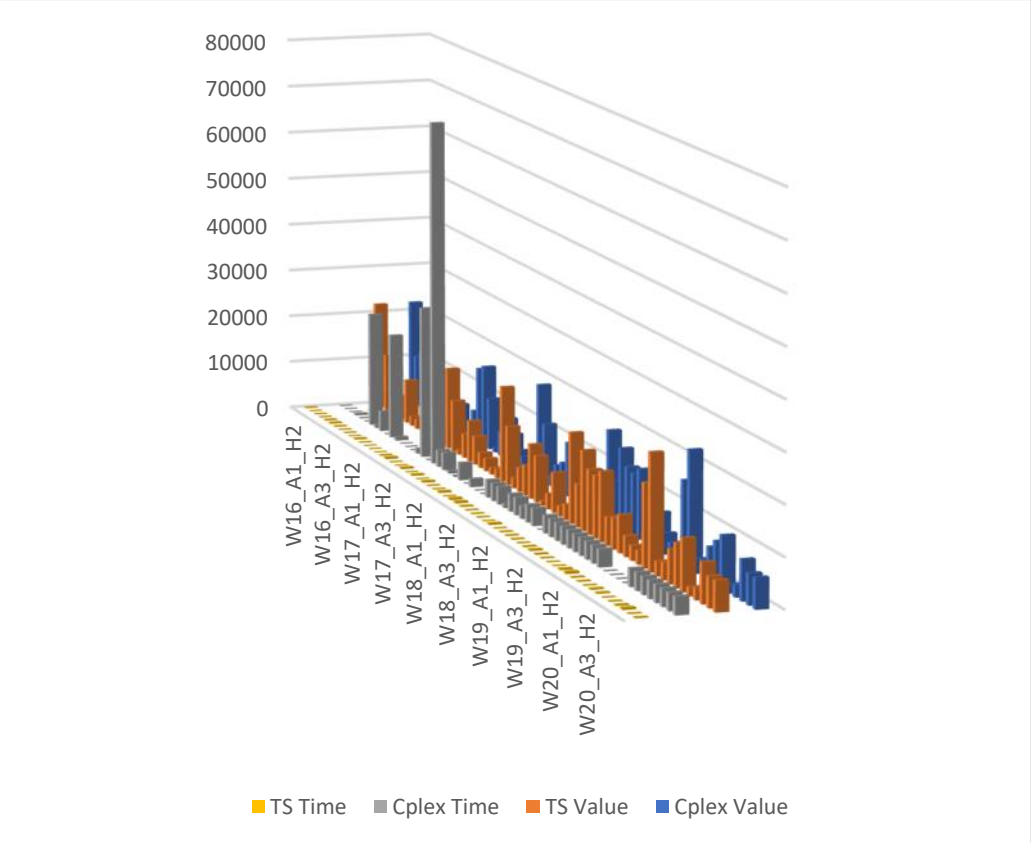


Figure 18. CPLEX vs. TS; value and time deviation

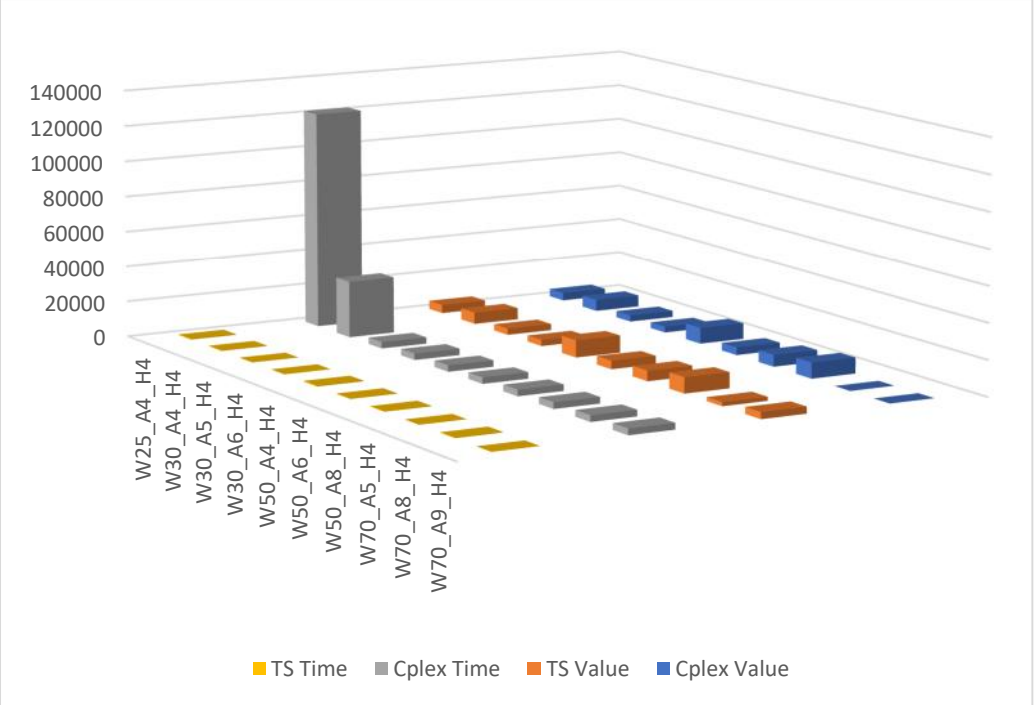


Figure 19. CPLEX vs. TS; value and time deviation - Big dimensions instances

After this unpromising result, we have adopted the hybridization between TS and SA on a small size instance with different characteristics as shown in the next table number 6. We applied (TS_SA) on just 5 instances; For 2 of them (W9_A7_H4) and (W10_A7_H4) TS_SA was able to provide optimal solutions in less 16 secs also the CPU consuming time was lower than the one consumed by TS alone. For the 3 remaining instances, TS_SA could not reach optimality even with more wasted CPU consuming time, deviation was between 8% and 27%, while with TS alone the deviation was varying between 23% and 42%. Note here that TS alone took more CPU consuming time than TS_SA for all of these five instances. We can clearly conclude that it was another failure in term of quality and processing time for this type of hybridization.

Table 6. CPLEX vs TS_SA

Instances	C.V	TS. V	TS_SA V	TS. CPU (sec)	C. CPU (sec)	TS_SA CPU (sec)	C.Gap	TS.Gap	TS_SA.Gap
W9_A7_H4	720	720	720	20	2	9	-	-	-
W10_A7_H4	720	720	720	22	25	15	-	-	-
W13_A7_H4	720	1020	780	31	21	826	-	41.7%	8.3%
W15_A7_H4	780	960	960	43	3282	122	-	23.1%	23.1%
W18_A7_H4	900	1260	1140	42	14994	175	7.14%	40.0%	26.7%

We visualize these results in figure 20.

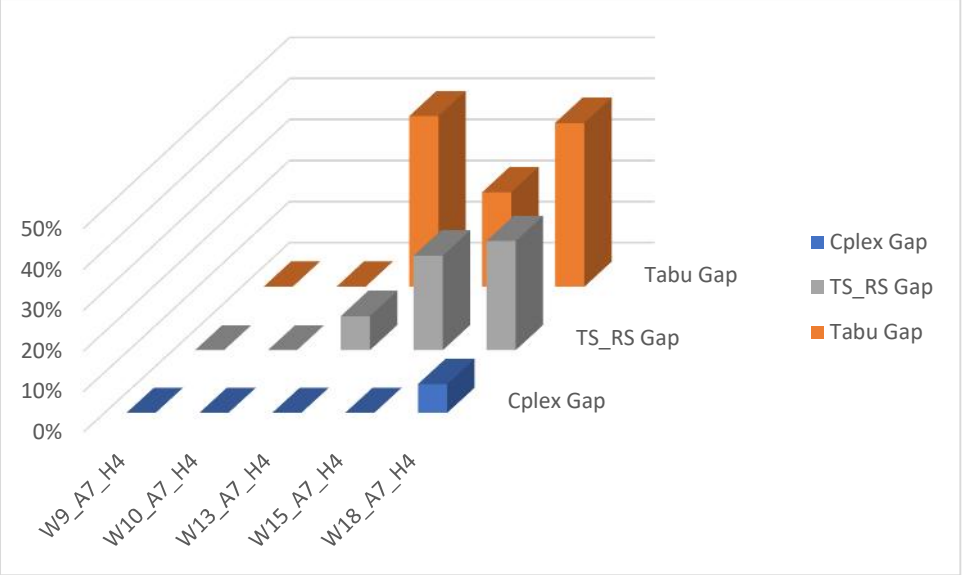


Figure 20. CPLEX - Tabu - TS_SA deviation

After these failures, we recognize that, for small size instances, TS improves results quickly but when we move to mid-size instances, it can easily stick in a local optimum and needs more time to escape from it. The hybridization with simulated annealing improves TS results but with more precious time wasted which make it useless due to time limitation which is a hard constraint to respect in order to achieve our objective.

3.7 Conclusion

In this chapter, we study a real-life case of wounded carrying toward hospitals via ambulances throughout a crisis in the case of sudden disaster. This Integrated Problem of Ambulance Scheduling and Resource Assignment (IPASRA) has been treated and formulated as a binary linear model that minimize the total time spent by routing ambulances picking up wounded from their locations to distant hospitals. We applied an effective hybrid meta-heuristic algorithm based on the Tabu search and (GRASP). We also offered a methodology that can be adopted to provide an efficient evacuation plans under various circumstances. Two scenarios were treated and for both of them, the offered model has proved his capability of evacuating wounded from their positions to hospitals within reasonable time

After discussing the obtained results for both scenarios in Section 3.5.3, and depending on the situation, we can notice the huge importance of choosing between the two scenarios especially with large instances, the choice should be based on the on-site provided information, especially wounded number and hospitals theoretical capacities, before taking the final decision. Both exact and hybrid versions of algorithms have been extensively tested on randomly generated instances of the (IPASRA). Our algorithms provide in both

scenarios' optimal solutions for instances with wounded number up to 50, ambulance number up to 8 and hospitals number up to 4. Also, it gives an efficient solution for instances with more than 70 wounded, 8 ambulances and 4 hospitals in less than 5 min of execution time, and even when it deals with an instance with 200 wounded, 15 ambulances and 5 hospitals or more, it continues providing feasible solutions in less than 10 min of CPU consuming time which prove its efficiency in term of quality and time.

At the end, the results of the proposed hybrid algorithm are compared with those obtained with the exact method of CPLEX solver. The comparison of the execution time illustrates the efficiency of our proposed model; its results show clearly that it is capable of generating optimal solutions within seconds. The CPLEX resolution, based on our modeling, also proves its efficiency but with more precious time wasted. However, the efficiency of the exact method decreases as the number of instances grows; it is only effective with a medium number of instances.

Chapter IV

A hybrid GRASP-Tabu Search approach to solve the integrated problem of ambulance missions' scheduling and resources assignment in the events of multiple disaster zones.

Chapter IV – A hybrid GRASP-Tabu Search approach to solve the integrated problem of ambulance missions' scheduling and resources assignment in the events of multiple disaster zones.

4.1 Motivation and Introduction

In this decade, many natural and triggered disasters have occurred all over the world and they have caused serious damages to infrastructure, death to thousands of people and they affect the economic development, such as Indonesia tsunami[x] in 2004. 5.12 Wenchuan earthquake[xi] in 2008, freezing rain disaster in southern China[xii] in 2008, devastating 2011 earthquake in Japan[xiii], flood disaster in India[xiv] in 2013 and hail disaster in Yancheng[xv] in 2016, Jiangsu. Especially in July 2016, heavy rain continues to strike the north of China[xvi], with severe flood in southern China at the same time. In addition, on the 7th of February 2009 a bushfire occurred in the state of Victoria[xvii] - Australia and causes lot of damages in properties and killed 174 citizens and around 414 wounded with different injuries

However, in addition to real disasters, the occurrence of terrorist attacks was growing up frequently all over the world. All of these attacks are treated as high-level crisis, which cannot be predicted. When it happens, we must act immediately and have the ability to make good and efficient decisions as fast as possible in order to minimize the human loss.

Citizen mass casualty events are becoming more and more important security issue all over the world since terrorism is not an endemic phenomenon, but an epidemic (Stein M, 1999) For example:

- **Oklahoma City attack[xviii]:** On April 19, 1995, The Alfred P. Murrah Federal Building in Oklahoma City was partially destroyed by a terrorist bomb. Injuries were sustained by 759 people, 168 of whom died. Most wounded were transported to local medical centers by private vehicles within 2 hours. More severely injured survivors were transported by ambulances.
- **Tripoli attack in Lebanon[xix]:** On the 23rd of august 2013 the terror conquered the city. Two mosques were hit by 2 cars bomb who have exploded after a lapse of 2 minutes of time, this caused the death of 87 citizens and more than 500 injured people with more than 150 in a critical condition.
- **Sinai attack in Egypt[xx]:** On November 24, 2017, a mosque in northern Sinai was attacked by an armed group that caused 305 deaths (including 28 children) and 120 injuries in a relatively small geographical area and far from hospitals.

Evacuation is one of the most important activities in disaster response. The purpose of this chapter is to propose a methodology that solve efficiently the (IPASRA). It determines which ambulance should serve which crisis zone and transport some or all of its wounded to a specific hospital. Our main target is to minimize the total travelling time of transporting ambulances between disaster zones and hospitals in order to reduce the mortality rate to the minimum. At first, we designed the integrated problem of ambulance missions scheduling and resources assignment (IPAMSRA) model as a binary linear program and then, in a subsequent step, we presented a mathematical modelling succeeded by a Branch and Cut resolution under CPLEX solver. Secondly, due to the computational complexity of such complicated problems, we proposed a hybrid algorithm that outperforms traditional metaheuristics methods in term of both resolution time and effective solution, this algorithm counts on the alternation between two algorithms: Tabu Search and (GRASP) procedure (GRASP-TS).

4.2 Mission

The purpose of this study is revealed by proposing an evacuation planning method that only depends on ambulances. Moreover, we propose a novel approach using hybridization between two metaheuristic methods Tabu Search and (GRASP) to deal with such complex cases where we have multiple dispersed disaster zones and multiple distant hospitals, our mission is to reduce the overall wounded's transport time from their physical locations to hospitals in order to reduce the overall mortality rate.

The aim of this chapter is to provide the best ambulance scheduling mission(s) and resources assignment process, it determines for each ambulance a single mission or a sequence of missions to follow. Each mission is characterized by a sequence of tasks to be performed where each task has its own predefined wounded number, starting and ending locations.

4.3 Problem description and mathematical modelling

This case of study is described as follow:

In large-scale crisis, we have many wounded grouped in several disasters' zones. We should transport them as fast as possible to hospitals. In fact, due to the large wounded number and to the hospitals limited capacities, we are forced to distribute wounded to several hospitals.

4.4 Problem description

We are concerned in studying inside a crisis management problem, a particular case of wounded evacuation where there are many groups of wounded and we have to transport *n* disaster zones to *m* hospitals sites with *k* ambulances (Figure 21). We deal with the Integrated Problem of Ambulance Missions Scheduling and Resources Assignment (IPAMSRA) where the resources to assign are hospitals and ambulances. The situation is described as follow:

We have a limited number of distant disaster zones, several ambulances located in one ambulatory center and several hospitals geographically distant from each other's. Each ambulance can serve several disaster zones located in its coverage area. We name the group of ambulance number, associated disaster zones, hospital numbers and wounded numbers as an entity, in other term; the whole transportation problem is divided into several sub problems where each sub-problem is an entity. Solving the whole problem is equivalent to solve one entity. The same strategy applies on all others entities.

We consider two different scenarios of the (IPAMSRA). In the first one, we treat the (IPAMSRA) using hospital capacity constraint and in the second one, we remove that constraint, in other term, we consider that all hospitals have an infinite capacity.

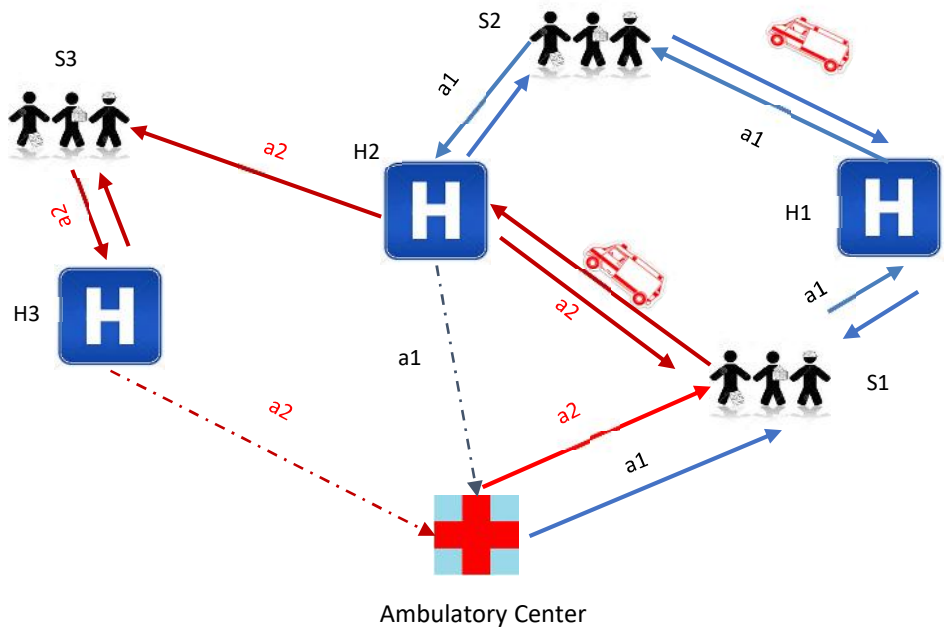


Figure 21. Ambulance-crisis zones- hospital assignment methodology

Assumption

Some assumptions should be respected.

- The total number of disaster zones, wounded number in each disaster zone and ambulances number are already known in advance.
- We will assume that the total number of ambulances to be used for wounded transportation purpose is less than the total number of wounded, also we will limit the ambulance capacity to one wounded at a time (Standard capacity).
- Each hospital h_i is able to serve a finite number w_i of wounded and has the ability to treat all kind of injuries.
- All disasters zones are distant from each other.
- We will not consider the wasted time spent by the operation of Loading/Unloading wounded from its location to hospital.
- We will assume that the best paths are always used by ambulances.
- Each disaster zone is considered like a round field with small radius and contains a finite number of wounded.

Generally, each wounded is associated to one medical diagnostic which is divided in four categories:

1. Critical injury
2. Severe injury
3. Moderate injury
4. Slight injury

When the injuries are classified as critical or severe, it means that we should act fast and provide help to wounded as fast as possible in order to save their lives. Usually, in these two cases the wounded maximum allowed waiting time is up to 30 minutes after that the injured will be most probably dead. To reduce mortality rate and due to the complexity of our proposed problem, we should adopt an efficient fast algorithm that is capable to solve such complex problems and can quickly provide efficient solutions within a small processing time which will not exceeds the 10 min barrier. To achieve our goal, we propose a hybrid GRASP-Tabu Search algorithm that will be detailed in section 4.6.

To prove that our problem is NP-Complete, we consider a specific instance where each sinister zone contains only one single wounded, in this case, our problem is similar to the Vehicle Routing Problem (V.R.P), which is known as Np-Complete problem.

4.5 *Mathematical Formulation*

- H = $\{H_1, H_2, \dots, H_m\}$ is the set of used Hospitals where |H| = m,
- S = $\{S_1, S_2, \dots, S_n\}$ is the set of Sinister zones where |S| = n,
- A = $\{A_1, A_2, \dots, A_k\}$ is the set of Ambulances where |A| = k,
- w_i: Theoretical capacity of hospital H_i, it is the maximum number of wounded that hospital H_i can handle at the beginning of the operation (i = 1, 2, …, m),
- $T_{(s,h)}$: Matrix used to specify the total time spent in transporting all wounded belonging to zone *s* toward hospital *h* ∀ (s, h) ∈ S × H,
- t_a: The time spent in accomplishing all of missions belonging to ambulance ***a*** ∀ a ∈ A,
- *G:* A sufficiently large number,
- P_s : Set of wounded in sinister zone ***s*** where |P| is the total wounded number in all disaster zones.

We define the following binary integer variables:

$x^a_{(s_1,h_1)(s_2,h_2)}$ =
- 1 if ambulance « ***a*** » serves zone s_1 towards hospital h_1 before serving zone s_2 towards hospital h_2
- 0 if else

$Z^a_{(s,h)}$ =
- 1 if ambulance « ***a*** » is affected to hospital h for serving zone ***s***
- 0 if else

Our main goal is to minimize the total time spent to serve all disaster zones.

C_M : The finish time of the last assigned vehicle ***a*** for serving the last group of wounded in the last disaster zone ***s***. In other term, the operation termination date named makespan.

Using the previous notation, solving our problem is equivalent to solve the following binary linear model:

Min C_M

Subject to:

$$\sum_{\substack{(s_2,h_2)\in S\times H \\ (s_2,h_2)\neq(s_1,h_1)}} x^a_{(s_1,h_1),(s_2,h_2)} \leq 1 \qquad \forall (s_1,h_1) \in S \times H;\ \forall a \in A; \qquad (1)$$

$$\sum_{\substack{(s_1,h_1)\in S\times H \\ (s_2,h_2)\neq(s_1,h_1)}} x^a_{(s_1,h_1),(s_2,h_2)} \leq 1 \qquad \forall (s_2,h_2) \in S \times H;\ \forall a \in A; \qquad (2)$$

$$\sum_{(s_2,h_2)\in S\times H} x^a_{(s_1,h_1),(s_2,h_2)} = \sum_{(s_2,h_2)\in S\times H} x^a_{(s_2,h_2).(s_1,h_1)} \quad \forall (s_1,h_1) \in S' \times H' \qquad (3)$$

$$\sum_{\substack{a\in V \\ s\in S}} Z^a_{(s,h)} \leq c(h) \qquad \forall h \in H \qquad (4)$$

$$\sum_{a\in V} Z^a_{(0,0)} = 1 \qquad \forall a \in A \qquad (5)$$

$$\sum_{\substack{a\in V \\ h\in H}} Z^a_{(s,h)} = |P(s)| \qquad \forall s \in S \qquad (6)$$

$$t_a=\sum_{\substack{(s_1,h_1)\in S*H \\ (s_2,h_2)\in S*H}} \left((2Z^a_{(s_1,h_1)} - \sum_{(s_2,h_2)\in S*H} x^a_{(s_1,h_1),(s_2,h_2)}\right) \times T_{(s_1,h_1)} + \sum_{,(s_2,h_2)} x^a_{(s_1,h_1),(s_2,h_2)} \times T_{(h_1 s_2,)}) \quad \forall a \in A \quad (7)$$

$$Z^a_{(s_1,h_1)} \leq \sum_{\substack{(s_2,h_2)\in S\times H \\ (s_2,h_2)\neq(s_1,h_1)}} x^a_{(s_1,h_1),(s_2,h_2)} \qquad \forall a \in A, \forall (s_1,h_1) \in S' \times H' \quad (8)$$

$$G.Z^a_{(s_1,h_1)} \geq \sum_{\substack{(s_2,h_2)\in S\times H \\ (s_2,h_2)\neq(s_1,h_1)}} x^a_{(s_1,h_1),(s_2,h_2)} \qquad \forall a \in A, \forall (s_1,h_1) \in S' \times H' \quad (9)$$

$$C_M \geq t_a \qquad \forall a \in A \quad (10)$$

Constraints (1) means that each mission (s_1, h_1) has at most one successor, constraint (2) ensure that each mission (s_1, h_1) admit at most one predecessor. Constraints (3) ensures the stream conservation, if a mission has a predecessor then it must admit a successor.

Constraint (4) guarantees that the total number of transported wounded to each hospital should not exceed its maximum capacity. Constraint (5) ensures that the first mission (0, 0) made by each ambulance had exactly one predecessor and one successor. Constraint (6) ensures that the total wounded number of each sinister zone should be transported to hospitals with the fleet of ambulances. Constraint number (7) evaluates the end of transporting time for all wounded served by an ambulance "a" after accomplishing all of its missions. Constraints (8) and (9) are binding constraints between variable A and variable x. Constraint (10) is used to evaluate the makespan which is the time spent by the last mission ending.

According to crisis degree, the problem can be considered with or without capacity constraint (4). Generally, in a major disaster, the number of injured is larger than the total hospitals capacities; in this case the capacity constraint should be ignored. Yet, according to situation especially when the total wounded number in all sinister zones is lower than the total capacity of all hospitals, it is more convenient to apply the capacity constraint in the modeling in order to propose an adequate solution.

4.6 Resolution methods

We describe in this section the adopted algorithms which are CPLEX-TS (C-TS) and (GRASP-TS), after that we discuss their hybridization and their adaption to our problem.

4.6.1 GRASP-TS

At first, we initialize some parameters such as the routing time between disaster zones, ambulances, hospitals and the best solution, after that, we apply (GRASP) procedure to construct a feasible solution. At each iteration, we treat the selected zone with some or all of its related wounded and we select randomly one of the nearest hospitals and one of the ambulances. Next, within the same iteration, we update the distances between the sinister zone and the selected ambulance, the hospital capacity and the number of remaining wounded inside that sinister zone. Then we move to the next

sinister zone. After we finish the treatment of all sinister zones, we check if the constructed solution is better than all of the previous constructed ones, in the affirmative case we save it as the best solution, after that we reinitialize all of the randomized procedure related parameters and relaunch it again and again until the predefined total iteration number is reached. At this point the algorithm is terminated and the best solution found is returned.

Before starting the Tabu Search procedure, we define a negative value D which will not affect the best solution previously found by (GRASP). The neighborhood construction will be detailed in the next subsection. At the beginning, all of the required parameters such as Tabu list size, current solution, best solution and the number of the allowed successive iterations where D remains negative are initialized. Then we start the iterations of our algorithm. In the beginning of the execution process, we define the maximum iterations number and we associate a negative neighbor value because it will not affect or improve the value of the best solution found. We use the number of successive unimproved iterations as an intensification/diversification criterion, and a stopping criterion. At each iteration, a neighborhood of the current solution is constructed. The number of successive unimproved iterations determines the number of the created neighborhoods at the current iteration. The actual solution is switched by the best neighbor's solution, which does not exist in the Tabu list. Moreover, in the case where the latter is better than the best-explored solution, it will be the best solution explored and it will be inserted in the Tabu list. When the maximal size of Tabu list is reached, the oldest element is removed. In addition, once the number of successive negative unimproved iterations is equal to a fixed integer denoted by I_M , the Tabu Search is terminated and the best solution is returned. Successively, the process of neighborhood construction is detailed.

The neighborhood of each solution created is divided into three different well-constructed sub neighborhoods. The adopted method in the first neighborhood construction consists in selecting a hospital at first then at second in replacing it with a new one and updating their current capacities without violating both hospitals capacities. For the second sub-neighborhood construction we select an ambulance and we apply the same strategy as the first one but only on that ambulance associated hospital, wounded and sinister zone and finally for the third sub neighborhood construction, we select a sinister and we only update the number of wounded to be transported without violating hospital capacity and with respect to the sinister cardinality.

4.6.2 Hybrid Algorithm

In this section, we show the adaptation of these two algorithms with our problem. Hybrid algorithms are popular and widely applied due to their efficiency compared to classic meta-heuristics. Hybrid methods have been widely relied on to solve complex problems. The main reason to use hybridization is to avoid falling in some problems especially the stagnation under local optima, also hybridization is characterized by its convergence speed to numerical result, and it reduces effectively the required processing time spent in calculating an optimal solution or an efficient solution. Yet, in many cases, metaheuristics encounters some convergence problems in solving problems to optimality. Recently, the cooperation structures between metaheuristics and exact approaches is becoming more and more adopted by researchers. By applying these strategies, we benefit from the advantages of both types of methods and enhance them to find efficient solutions.

In literature, we have found many hybridization strategies but we are interested in just two types of them. The first type of hybridization counts on using even parallelization or threading, its technique depends on the exploitation of concurrence and benefits from the cooperation between many meta-heuristic processes that are launched in parallel. The second type is the sequential hybridization, its technique counts on the alternans between two or more meta-heuristics and on the cooperation between them such as the communication among some parameters. The strategy of the offered hybrid algorithms consists in a sequential hybridization of (CPLEX) algorithm and (TS) procedure (C-TS) for the first one and a sequential hybridization of (GRASP) algorithm and (TS) procedure (GRASP-TS) for the second one. This (GRASP-TS) approach is based on restarting the Tabu Search from different initial solutions previously generated by (GRASP) algorithm, while the (C-TS) consists on launching TS procedure from the initial solution constructed with CPLEX.

We describe our hybrid (GRASP-TS) algorithm with the integrated Tabu Search procedure in the following figures 22 and 23.

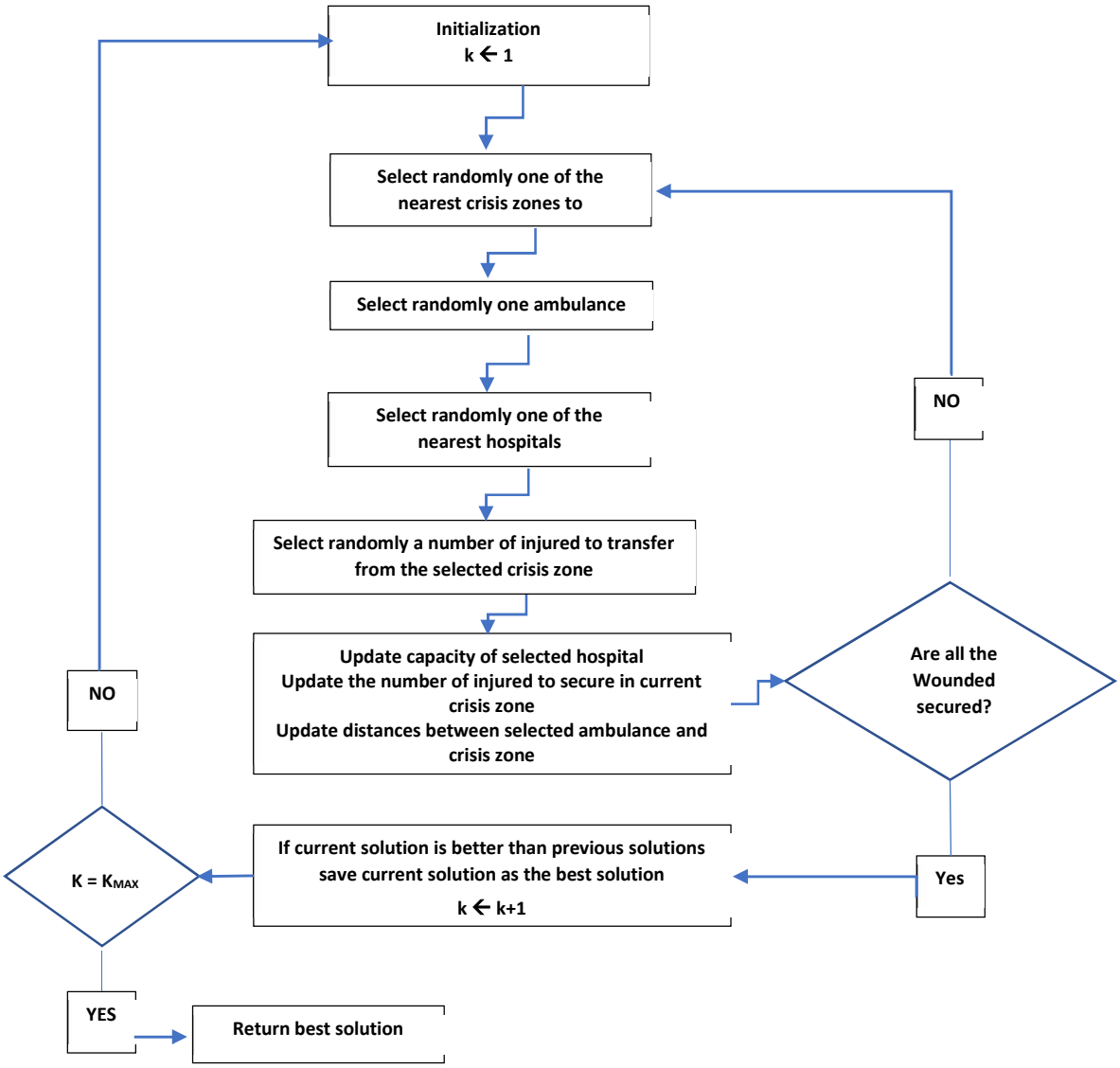


Figure 22. GRASP procedure

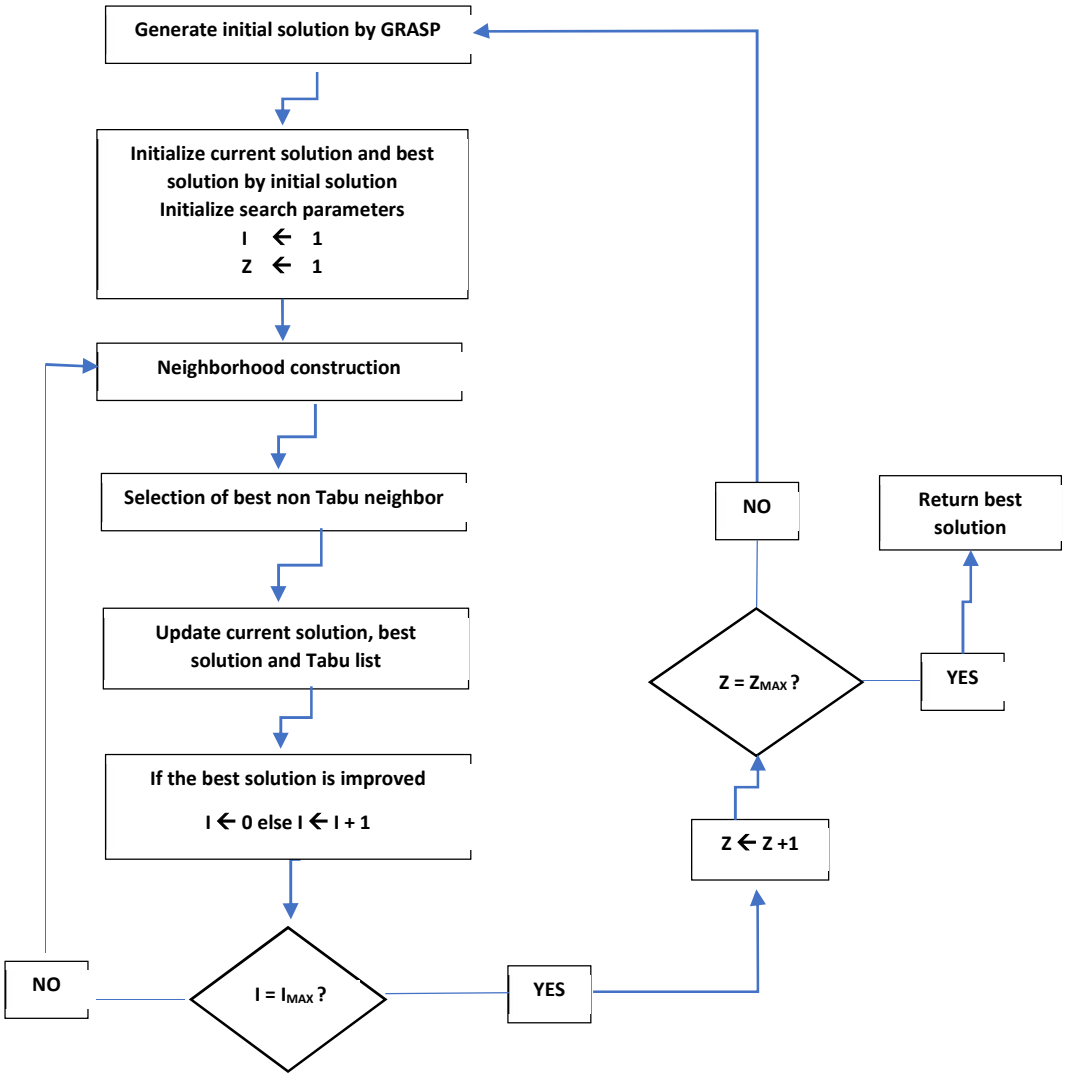


Figure 23. GRASP_TS procedure

Note here that the values of I_(Max),K_(Max) and Z_(Max)are subject to modifications depending on the instances size. We present in the next table 7, the values of theses iterations according to instances size.

Table 7. Parameter's values

Family	K_M	I_M	Z_M
F1	10000	1000	1000
F2	500000	2000	3000
F3	1000000	3000	5000

4.7 *Experimental results and Discussions*

All algorithms are coded with JAVA-NetBeans 8.2 programming language on a Windows 8.1 pro using Toshiba Satellite Core I7, 2.4 GHZ and 16 GB of RAM with 500 GB SSD Hard disk and 2GB AMD RADEON dedicated VGA. These results are compared with those obtained by CPLEX V12.71 running on the same machine.

Instances are divided in three families F1, F2 and F3. F1 contains instances with a number of injured variants between 50 and 200 wounded, ambulances variant between 5 and 20, hospitals variant between 2 and 5, sinister variant between 2 and 4. F2 contains instances of 300 to 450 wounded ambulances variant between 30 and 40, hospitals variant between 6 and 8, sinister variant between 6 and 12 and F3 is the family that contains instances of more than 400 wounded, 40 ambulances, 8 hospitals and more than 15. The deviation between CPLEX solution and the solution given by hybrid algorithm is calculated by:

$$\text{Gap} = \frac{V.H.A. - V.C.}{V.C.}$$

V.H.A.: Value given by Hybrid Algorithm

V.C.: Value given by CPLEX

Starting with the proposed hybrid algorithm (GRASP-TS), for F1 family, we limit the CPU execution time for 3 min maximum then for family F2, the processing time is limited to 7 minutes' maximum and finally when we move to the third family F3, the time barrier is fixed to 15 minutes' maximum.

Meanwhile, for CPLEX, we limit the CPU consuming time for 1 hour for small and mid-size instances (Families F1 and F2), but when we move to the big size instances (Family F3), the execution time is fixed for a maximum of 3 hours of continuous execution.

In this paper, we compute the consumed time spent by each ambulance from the beginning of its missions until their end. The time that we intend to reduce is the total time spent by the latest ambulance mission.

Detailed results are given in two tables and graphically shown in four flow charts.

- First scenario: Instances with hospitals capacities
  - Tables 9.
- Second scenario: Instances without hospitals capacities
  - Tables 10.

For tables 9 and 10 we define the following properties

- **Family description:** for example, Instance name Pd_Ak_Hm_Sn means that the total number of wounded P is equal to *d*, the number of used ambulances A is equal to *k,* the total number of hospitals H in use is equal to *m* and finally we determine the total number of sinister zones S that is equal to *n*.
- **Columns description:** The first column shows the Family name; the second one shows the instance size. The third fourth and fifth ones show the obtained objective value in seconds with CPLEX and hybrids algorithms respectively, the sixth, seventh and eighth columns show the consuming CPU runtime, in seconds, for

CPLEX and hybrids respectively and finally we show the calculated gap for CPLEX and hybrid in columns 9, 10 and 11 respectively.

To test the performance of (GRASP-TS) we implement different instances with large problem sizes, results of large instances with same number of hospitals are treated.

Note here that when we intend to use hospitals with capacities, we assume that all hospitals capacities can cover the total number of wounded coming from all sinister zones. In addition, we should mention here that the distribution of wounded among disasters zones is totally randomized.

In table 8 we determine for each family, the hospitals in service and the adopted capacity for each one of them.

Table 8. Hospitals available capacities

Family	H1	H2	H3	H4	H5	H6	H7
F1	42	48	35	50	95	-	-
F2	42	48	35	50	95	365	280
F3	142	148	135	150	295	265	280

Note here that the deviation between CPLEX solutions and hybrid solutions was measured; when negative it means that the hybrid generated solution better than the best known one generated by CPLEX due to memory or processing time limitation.

Tables 9 and 10 shows results for a sample of the generated instances (F1, F2 and F3) used for simulation assessments. Note that at the beginning of the secure operation, we provide for each instance three other data: The distances between hospitals and sinister zones, the distances between ambulances and sinister zones and the wounded number in each disaster zone.

For F1 family we limit CPLEX CPU consuming time to 3600 secs and for F2 and F3 Families, the CPU consuming time was limited to 10800 secs.

In table 9. We examine three families of instances F1, F2 and F3. From twenty-seven instances CPLEX provides optimal solutions for just three of them in less than 1 sec, the worst gap calculated for the remaining instances was equal to 10.04% in instance (W400_A40_H7_S3) and the best gap attained 0.2% in instance (W200_A15_H5_S4).

To evaluate the solution quality for the hybrid algorithm (GRASP-TS), we realize that, for F1 family, it produced for instance (W200_A15_H5_S4) the same value provided by CPLEX in 1 minute of CPU processing time. For the remaining instances the processing time is limited to 3600 secs, the deviation between CPLEX produced solutions and solutions given

by (GRASP-TS) varies between (0.14%) with instance (W125_A15_H4_S3) and (3.99%) with instance (W150_A15_H4_S3). For F2 family, the processing time is limited to 10800 secs, none of the instances produced by CPLEX were optimal. The deviation between these solutions and solutions given by hybrid algorithm is between -1.33% with instance (W400_A40_H7_S3) and 8.69% with instance (W500_A50_H7_S7). Expecting instances (W450_A40_H7_S5), (W500_A50_H7_S7) and (W550_A50_H7_S7), the hybrid algorithm calculated gap is less than 6 %. The calculated gap for those instances were respectively 6.83%, 8.69% and 8.87%.

Hence, for F3 family, the CPLEX gaps for instances (W600_A35_H6_S3), (W600_A50_H7_S3), (W700_A45_H6_S5), (W900_A50_H6_S4), (W900_A50_H7_S7), (W1000_A45_H6_S5) and (W1000_A50_H7_S7) are respectively 3.32%, 2.11%, 3.99%, 3.51%, 6.8%, 7.52% and 5.15%. However, for instance (W800_A40_H7_S3) CPLEX was unable to deliver feasible solution after 10800 secs of CPU consuming time, it returns an "out of memory" message. In the meantime, and for the same instances excepting (W800_A40_H7_S3), our hybrid algorithm provides efficient solutions with a gap varying between (2.11%) and (7.52%) in less than or equal to 10 minutes of CPU consuming time. Moreover, it continues providing feasible solution for instance (W800_A40_H7_S3) and beyond in less than 10 minutes of CPU consuming time.

Note here that when the deviation between CPLEX solutions and hybrid solutions is negative, it means that the hybrid generated solution is better than the best known one generated by CPLEX due to memory or processing time limitation.

Table 9. CPLEX vs C.TS vs GRASP-TS with capacity constraint.

	Instance ID	*C V (sec)*	*GRASP-TS V (sec)*	*C CPU (sec)*	*GRASP-TS CPU (sec)*	*C GAP*	*GRASP-TS GAP*
	W50_A5_H2_S2	10124	10146	<1	31	0.00%	0.22%
	W75_A5_H2_S2	8393	8489	<1	36	0.00%	1.14%
	W100_A10_H3_S2	3659	3714	<1	43	0.00%	1.50%
	W125_A10_H4_S3	5825	6009	3600	294	0.23%	3.16%
F1	W125_A15_H4_S3	4371	4377	3600	38	2.21%	0.14%
	W150_A10_H4_S3	9537	9829	3600	550	0.36%	3.06%
	W150_A15_H4_S3	8395	8730	3600	350	0.33%	3.99%
	W200_A15_H5_S4	8878	8878	3600	61	0.20%	0.00%
	W200_A20_H5_S4	8906	9072	3600	233	2.08%	1.86%
	W300_A35_H6_S2	3458	3501	10800	289	5.23%	1.24%
	W300_A30_H6_S6	4296	4513	10800	276	2.47%	5.05%
	W350_A35_H6_S2	3377	3483	10800	600	0.35%	3.14%
	W400_A35_H6_S3	5767	5960	10800	327	1.27%	3.35%
F2	W400_A40_H7_S3	2400	2368	10800	503	10.04%	-1.33%
	W450_A40_H6_S4	8775	9126	10800	600	2.68%	4.00%
	W450_A40_H7_S5	5238	5596	10800	519	3.57%	6.83%
	W500_A50_H7_S7	4478	4867	10800	600	3.82%	8.69%
	W550_A45_H7_S6	4908	5179	10800	600	8.96%	5.52%
	W550_A50_H7_S7	6529	7108	10800	600	3.09%	8.87%
	W600_A35_H6_S3	7803	8062	10800	600	0.86%	3.32%
	W600_A50_H7_S3	8003	8172	10800	600	3.79%	2.11%
	W700_A45_H6_S5	2434	2531	10800	600	0.45%	3.99%
	W800_A40_H7_S3	N.S - O.M	11539	3413	600		
F3	W900_A50_H6_S4	9099	9418	10800	600	2.59%	3.51%
	W900_A50_H7_S7	3557	3799	10800	600	7.20%	6.80%
	W1000_A45_H6_S5	5170	5559	10800	600	1.95%	7.52%
	W1000_A50_H7_S7	11187	11763	10800	600	2.37%	5.15%

In the next table 10, we extant for the same instances, the results of the non-capacitated version of (IPAMSRA).

From twenty-seven instances, CPLEX was able to provide optimal solutions for six instances: (W50_A5_H2_S2), (W75_A5_H2_S2), (W100_A10_H3_S2), (W150_A10_H4_S3), (W300_A35_H6_S2) and (W350_A35_H6_S2) in a reasonable time less than 4 minutes. The Gap for all remaining instances varies between (0.77%) for instance (W600_A35_H6_S3) and (11.16%) for instance (W900_A50_H7_S7).

Regarding our hybrid algorithm, the calculated gap in the worst case was (8.78%) for instance (W500_A50_H7_S7), and (-1.11%) for instance (W900_A50_H7_S7) in the best case.

Starting with instances of F1family, the CPLEX gap is between (0.91%) and (7.39%), 4 of 9 instances were optimal, 3 of them were given in less than 1 minute and the remaining one was given in less than 3 minutes, instances (W125_A10_H4_S3), (W125_A15_H4_S3), (W150_A15_H4_S3), (W200_A15_H5_S4) and (W200_A20_H5_S4) did not attain optimality after a fixed processing time of 1 hour with a gap equals to (0.99%), (7.39%), (0.91%), (1.07%) and (1.7%) respectively. With the hybrid algorithm, from a total of 9 instances, none was optimal. For instance, (W125_A15_H4_S3) the hybrid algorithm produces a lower value than CPLEX with a gap of (-0.83%) in less than 3 minutes. For the remaining instances (W50_A5_H2_S2), (W75_A5_H2_S2), (W100_A10_H3_S2), (W125_A10_H4_S3), (W125_A15_H4_S3), (W150_A15_H4_S3), (W200_A15_H5_S4) and (W200_A20_H5_S4) the deviation given by the hybrid algorithm was equal to (1.79%), (2.71%), (1.84%), (3.12%), (4.07%), (3.73%), (4.76%) and (4.57%) with a processing time less than 250 secs.

Continuing with F2 family, instance (W300_A35_H6_S2) and (W350_A35_H6_S2), were solved optimally by CPLEX in less than 127 secs and the deviation between CPLEX solution and the solution given by hybrid algorithm is equal to (2.69%) and (0.7%) correspondingly. For all instances except for instances (W500_A50_H7_S7), (W550_A45_H7_S6) and (W550_A50_H7_S7) where the deviation is equal to (8.78%), (7.44%) and (8.15%), were given in less than 7 minutes. For instances (W300_A30_H6_S6), (W400_A35_H6_S3), (W400_A40_H7_S3), (W450_A40_H6_S4) and (W450_A40_H7_5) the CPLEX gap is between (2.26%) and (8.79%). The deviation between these solutions and solutions given by hybrid algorithm is between (1.62%) and (5.92%).

Ending with F3 Family, from a total of 8 instances, six feasible solutions were provided by CPLEX. The CPLEX gap, in instances (W600_A35_H6_S3), (W600_A50_H7_S3), (W700_A45_H6_S5), (W800_A40_H7_S3), (W900_A50_H6_S4) and (W900_A50_H7_S7) were equal to (0.77%), (5.6%), (3.99%), (0.98%), (3.12%) and (11.16%) respectively. For these instances (GRASP-TS) provides efficient solution with gap values between (-1.11%) and (7.11%) in less than 10 minutes except for instance (W700_A45_H6_S5) which is given in exactly 10 minutes. For the remaining instances (W1000_A45_H6_S5) and (W1000_A50_H7_S7) “No Solution, Out-Of-Memory” message was returned by CPLEX after 10800 secs of CPU processing time. The (GRASP-TS) continues generating feasible solutions in a reasonable time which can clearly demonstrate that for big instances (GRASP-TS) was more efficient than CPLEX in term of quality and CPU processing time.

Table 10. CPLEX Vs C.TS Vs GRASP-TS, without capacity constraint.

	Instance ID	*C V (sec)*	*GRASP-TS V (sec)*	*C CPU (sec)*	*GRASP-TS CPU (sec)*	*C GAP*	*GRASP-TS GAP*
	W50_A5_H2_S2	9968	10146	<1	25	0.00%	1.79%
	W75_A5_H2_S2	6044	6208	<1	37	0.00%	2.71%
	W100_A10_H3_S2	2874	2927	222	57	0.00%	1.84%
	W125_A10_H4_S3	5617	5792	3600	207	0.99%	3.12%
F1	W125_A15_H4_S3	1799	1784	3600	156	7.39%	-0.83%
	W150_A10_H4_S3	6362	6621	102.13	41	0.00%	4.07%
	W150_A15_H4_S3	7021	7283	3600	249	0.91%	3.73%
	W200_A15_H5_S4	3109	3257	3600	216	1.07%	4.76%
	W200_A20_H5_S4	8138	8510	3600	124	1.70%	4.57%
	W300_A35_H6_S2	1115	1145	4.14	148	0.00%	2.69%
	W300_A30_H6_S6	4186	4378	10800	228	2.63%	4.59%
	W350_A35_H6_S2	2000	2014	126	116	0.00%	0.70%
	W400_A35_H6_S3	2594	2636	10800	332	8.79%	1.62%
F2	W400_A40_H7_S3	1402	1485	10800	365	3.31%	5.92%
	W450_A40_H6_S4	7738	8046	10800	412	2.26%	3.98%
	W450_A40_H7_5	3906	4102	10800	396	4.87%	5.02%
	W500_A50_H7_S7	2084	2267	10800	600	8.54%	8.78%
	W550_A45_H7_S6	3080	3309	10800	552	2.27%	7.44%
	W550_A50_H7_S7	4748	5135	10800	600	5.33%	8.15%
	W600_A35_H6_S3	9037	9368	10800	447	0.77%	3.66%
	W600_A50_H7_S3	2926	3133	10800	501	5.60%	7.07%
	W700_A45_H6_S5	7571	8109	10800	600	3.99%	7.11%
	W800_A40_H7_S3	9236	9658	10800	533	0.98%	4.57%
F3	W900_A50_H6_S4	6790	7186	10800	575	3.12%	5.83%
	W900_A50_H7_S7	3162	3127	10800	568	11.16%	-1.11%
	W1000_A45_H6_S5	N. S-O.M	4851	1640.45	600		
	W1000_A50_H7_S7	N.S-O.M	4488	<1	600		

Detailed results are graphically shown in four flow charts.

- First scenario: Instances with hospitals capacities
  - Figures 24 and 25.
- Second scenario: Instances without hospitals capacities
  - Figures 26 and 27.

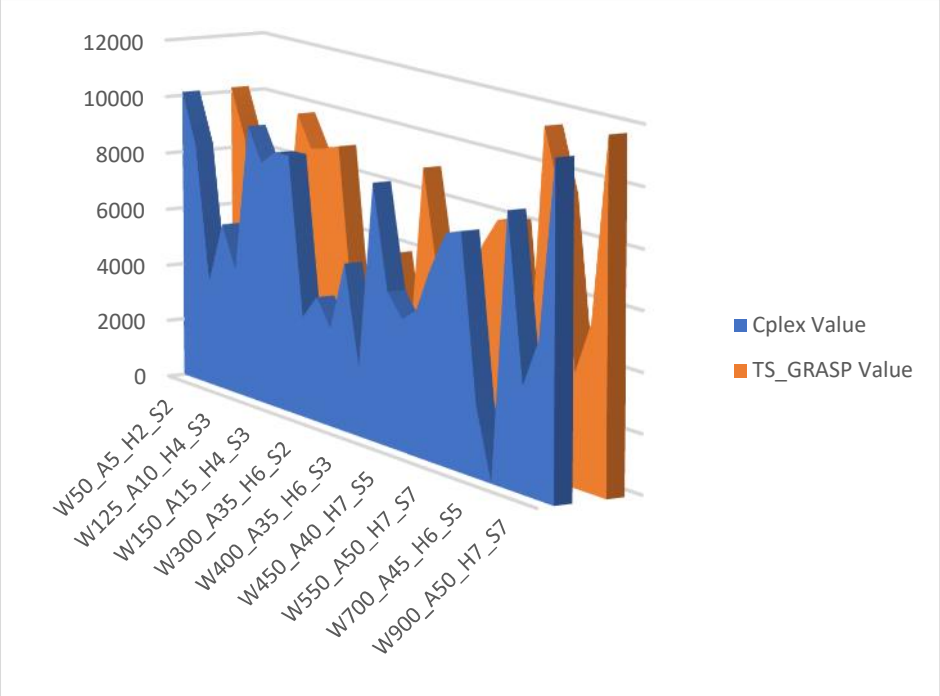


Figure 24. CPLEX Vs GRASP-TS, visual results with capacity constraint.

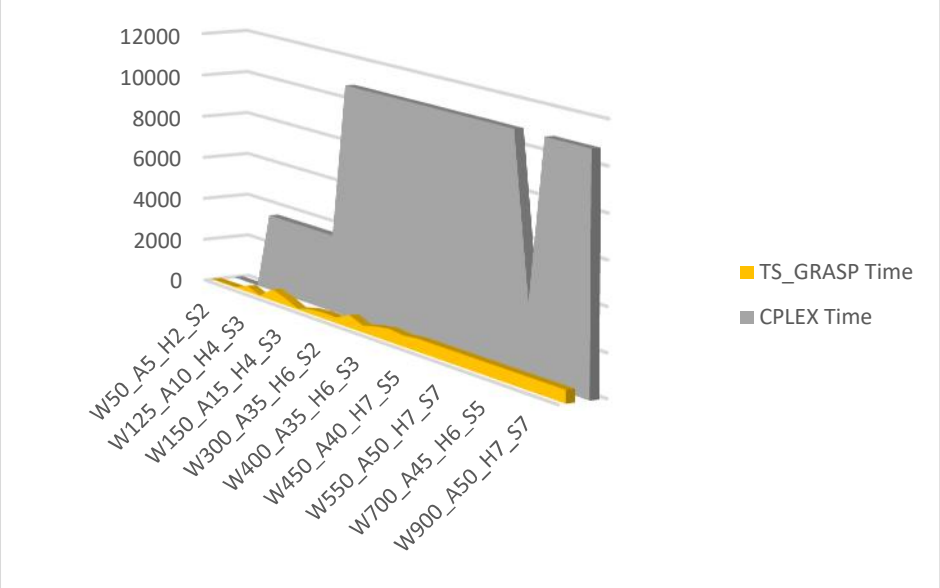


Figure 25. CPLEX Vs GRASP-TS CPU consuming time comparison with capacity constraint.

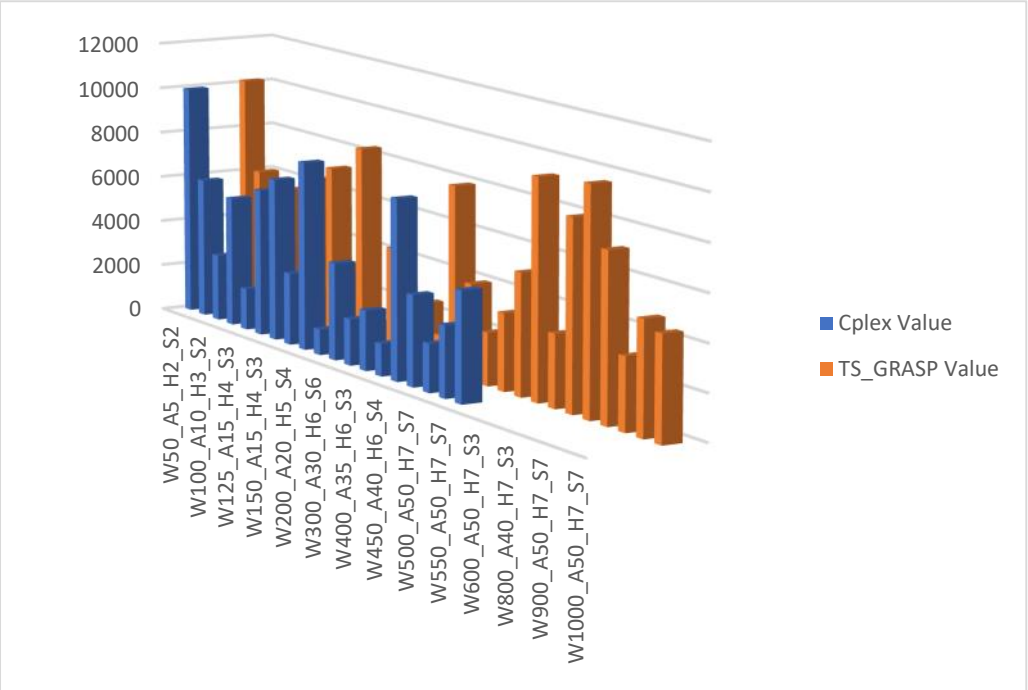


Figure 26. CPLEX Vs GRASP-TS values comparison (Capacity constraint excluded)

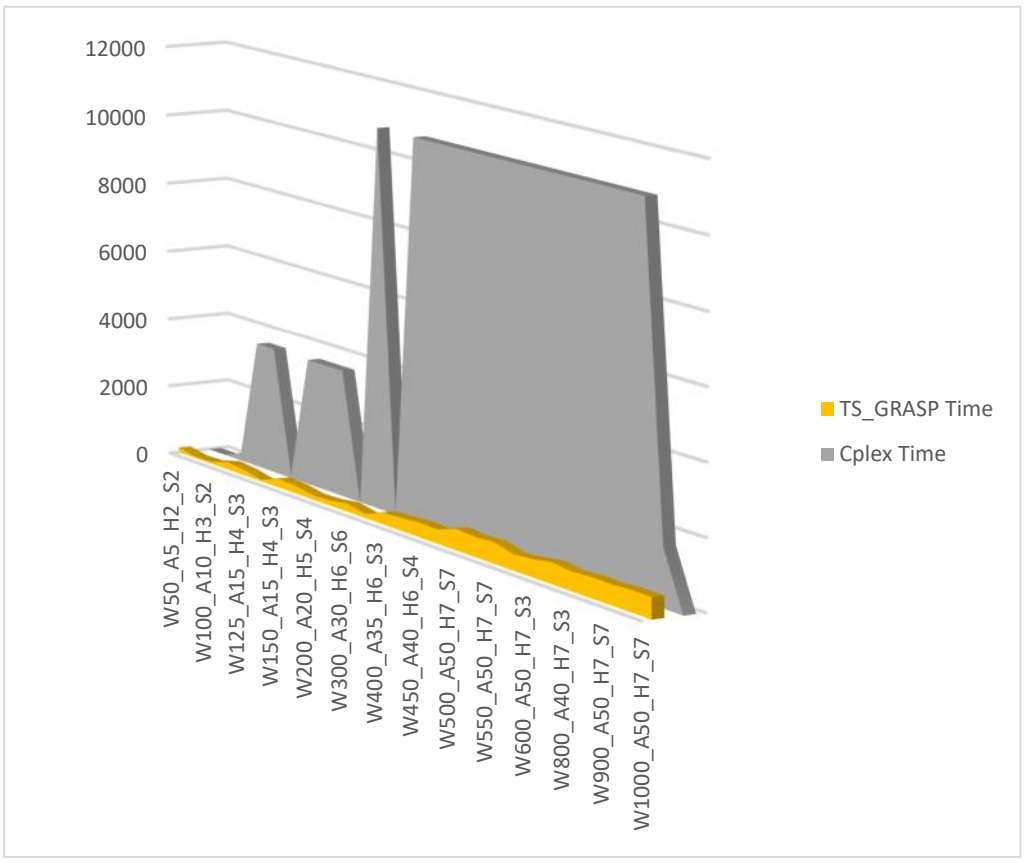


Figure 27.CPLEX Vs GRASP-TS processing time comparison (Capacity constraint excluded)

4.8 Conclusion

This chapter investigates optimization models for emergency evacuation plans in the case of sudden disaster. It develops a binary linear model and a hybrid approach to solve the (IPAMSRA) that minimize the total time spent by routing ambulances picking up wounded from their locations to distant hospitals in the event of multi disaster zones evacuation problem in an emergency response. A mathematical model and a hybrid (GRASP-TS) approach were applied to numerous situations with a set of randomly generated instances under various circumstances to examine their efficiency. The mathematical model was solved using CPLEX solver in order to evaluate the quality of solutions produced by hybrid (GRASP-TS). Our main concern is to minimize the total transporting time of wounded coming from multiple disaster zones to the designated hospitals sites. Two scenarios were treated. For both of them, the model has proved its capability of evacuating wounded within reasonable time. Another proof of its efficiency is its high speed in generating efficient solutions.

By solving our model using the exact method CPLEX, we can validate the results generated by our hybrid algorithms.

Both exact and hybrid versions of algorithms have been extensively tested on randomly generated instances of the (IPAMSRA). With F1 and F2 families, our (GRASP-TS) algorithm provide in both scenarios' efficient solutions in a reasonable processing time. Ending with F3 family, CPLEX solver provides feasible solutions for some instances in 10800 secs, but for the remaining instances, and due to Memory or processing time limitation, no solutions were provided. Meanwhile (GRASP-TS) continues providing, in both scenarios, efficient solutions in less than or equal to 10 min of CPU consuming time, which prove its efficiency in term of quality and processing time.

After discussing the generated results for both scenarios, and based on the situation, we can notice the huge importance of choosing between these two scenarios especially when it comes to large instances size. The choice should be based on the on-site provided information, especially wounded number and hospitals theoretical capacities, before taking the final decision. Our algorithm (GRASP-TS) provides in both scenarios' efficient solutions for instances, with a wounded number exceeding 1000, more than 50 ambulances number, over 7 hospitals and sinister number beyond 7 in less than 10 min of CPU consuming time.

At the end, the achieved results of the proposed algorithm were compared with those produced by the exact method of CPLEX solver. The evaluation of the completing time illustrates the efficiency of our proposed model for big size instances. Based on its results, we can confirm that it is useful and able to generate efficient solutions within a short time.

Regarding CPLEX, it is efficient only with small instances due to the mass resolution time that it takes. Accordingly, our proposed hybrid algorithm (GRASP-TS) is robust for small and mid-size instances and more suitable in term of quality and execution time for big size instances.

Conclusion (Second PART)

In this part, we present three different approaches to deal with wounded evacuation within crisis management, in addition to two different scenarios for the first two approaches.

In the same occasion, we show some failure applied tests for some of existing metaheuristics algorithms used to solve the wounded transportation problem.

Publications

- KHORBATLY M, DKHIL H, ALABBOUD H, YASSINE A., A hybrid algorithm Tabu Search – GRASP for wounded evacuation in disaster response, accepted for RAIRO October 2018.
- KHORBATLY M, DKHIL H, ALABBOUD H, YASSINE A., A binary linear modelling and a hybrid GRASP-Tabu Search approach to solve the integrated problem of ambulance mission scheduling and resources assignment for evacuation purpose in the events of multiple disaster zones, submitted to JORS, October 2018.

Conclusion

Conclusion

In the intervention of a sudden crisis or disaster, casualty evacuation plays a critical role in affecting, directing and orienting all rescue procedures and operations.

Our research work fits into the vast field of digital optimization and operational research. Within a crisis management logistic chain, this thesis requires the contribution of several optimization problems such the Vehicle Routing Problem (VRP) with capacity (CVRP) and the vehicles scheduling problem (VSP). These two problems are classified in the theory of complexity as NP-hard.

The scientific approach of our study is based on the combination between (C.V.R.P.) and (V.S.P). We investigated the Integrated Vehicle Sequencing and Resource Allocation Problem (IPASRA) to address the problem of transporting the wounded.

At the beginning, we started in the first part of the thesis by a state of the art on the supply chain, the different resolution strategies cited in literature and the conceptual framework of our research, as well as the most popular algorithms used for the resolution of our optimization problem, then we justified our choice of the adopted resolution methods whether at the conceptual level or in the application of research methods already mentioned.

Then, in the second part of our thesis, we introduced the integrated problem of Ambulance Scheduling and Resources Assignment to wounded (IPASRA), after that, we created instances inspired by real cases (Tripoli attacks in 2012, Paris attacks in 2015, and also of many other crises around the world).

In a first place, we have developed two mathematical models adapted to the different temporary situations already built. In a second place, we proposed hybrid resolution algorithms that are both fast and efficient. In the third chapter, we studied the (IPASRA) in the case where the wounded are separated and reside in several distant locations and we will transport them to several hospitals as far away as possible, then in the fourth chapter we generalized the (IPASRA) and introduced an integrated scheduling problem for ambulance missions and resource allocation (IPAMSRA). This problem is applied in cases where the wounded are grouped together in several remote areas and, at each ambulance is assigned a set of sub-groups of wounded transport missions to a target hospital located in distant areas, and so continued until all wounded, in all affected areas, are secured as soon as possible. We study two real resolution scenarios: the first scenario considers hospitals with limited casualty service capacity and the second scenario eliminates this constraint and considers that all hospitals must receive the total number of injured who arrive there. The choice between these two scenarios is based on the information received from outside agents on site, this information must specify the number of wounded, their physical

locations and the number of hospitals with their capacities, when the number of wounded exceeds the capacity of all combined hospitals, the second scenario is adopted.

Finally, we presented in the fifth chapter, a third approach in the process of development for the resolution of our problem where we integrate the principle of point cloud labelling in our research methodology. At the beginning, we explain the wounded grouping criteria, then, we pass to the numerical resolution always being based on the hybridization between (GRASP) and (TS) but with adaptation of the program already offered.

Perspectives

In this thesis, we have already started working with a new resolution approach that depends on creating virtual groups of wounded before starting the evacuation process.

For future work, we believe that there is a need for further improvements by:

1. Developing a meta-heuristic approach to solve the multi priority groups.
2. Adding dynamic hospital capacity.
3. Implementing the use of firefighter truck.
4. Implementing the notion of wounded urgency and transportation priority.

Conclusion

Dans l'intervention d'une crise ou d'un désastre soudain, l'évacuation de blessés joue un rôle essentiel qui affecte, dirige et oriente toutes les procédures et opérations de secours.

Notre travail de recherche s'encadre dans le domaine vaste de l'optimisation numérique et la recherche opérationnelle. Au sein d'une chaine logistique de gestion de crise, cette thèse nécessite la contribution de plusieurs problèmes de recherche d'optimalité surtout le problème de tournées de véhicules (V.R.P) en particulier avec capacité (C.V.R.P) et le problème d'ordonnancement de véhicules (V.S.P). Ces deux problèmes sont classifiés dans la théorie de la complexité comme NP-hard.

La démarche scientifique de notre étude est basée sur la combinaison entre (C.V.R.P) et (V.S.P). Nous avons étudié le problème intégré d'ordonnancement de véhicules et d'allocation de ressources (IPASRA) pour assurer la résolution du problème transport de blessés.

Tout d'abord, nous avons commencé dans la première partie de la thèse par un état de l'art sur la chaine logistique et les différentes stratégies de résolution et le cadre conceptuel de notre recherche, ainsi que les algorithmes les plus populaires utilisés pour la résolution de notre problème d'optimisation, puis on a justifié notre choix de méthodes de résolution que ce soit au niveau conceptuel ou dans l'application de méthodes de recherches déjà citées.

Ensuite, dans la seconde partie de notre thèse, on a introduit le problème intégré d'ordonnancement des véhicules et d'allocation des ressources aux blessés (IPASRA), puis on a créé des instances inspirées des cas réels (les attaques de paris en 2015, les attaques de Tripoli – Liban en 2012 et aussi de beaucoup d'autre crises autour du monde).

Dans un premier lieu, nous avons développés deux modèles mathématiques adaptés aux différentes situations provisoires déjà construites. Dans un second lieu, nous avons proposé des algorithmes de résolutions hybrides qui sont à la fois rapides et efficaces. Dans le troisième chapitre, on a étudié le (IPASRA) dans le cas où les blessés sont séparés et résident dans plusieurs locations distantes et on va les transporter vers plusieurs hôpitaux aussi distants le plus vite possible, puis dans le quatrième chapitre on a généralisé le (IPASRA) et on a introduit un problème intégré d'ordonnancement des missions des ambulances et d'allocation des ressources (IPAMSRA). Ce problème est appliqué dans le cas où les blessés sont regroupés ensembles dans plusieurs zones distantes et, à chaque ambulance est affecté un ensemble de missions de transport de sous-groupes de blessés vers un hôpital cible situé dans une location aussi distante, et ainsi de suite jusqu'à ce que tous les blessés, dans toutes les zones sinistrées, sont transportés le plus tôt possible. Deux scénarios réels de résolution

se présentent : le premier scénario considère que les hôpitaux possèdent de capacité de service de blessés limité et le second scénario élimine cette contrainte et considère que tous les hôpitaux doivent recevoir tout le nombre de blessés qui y en arrivent. Le choix entre ces deux scénarios se base sur les informations reçus des agents extérieurs en site, ces informations doivent préciser le nombre de blessés et leurs locations physiques et le nombre des hôpitaux avec leurs capacités, quand le nombre de blessés dépasse la capacité de tous les hôpitaux réunis, on a recours à adopter le second scénario.

Enfin, nous présentons, dans le cinquième chapitre, une troisième approche en cour de développement pour la résolution de notre problème où on intègre le principe de nuage de points dans notre méthodologie de recherche. Au début, nous expliquons les critères de groupage ensuite, on passe à la résolution numérique en se basant toujours sur l'hybridation entre (GRASP) et (TS) mais avec adaptation du programme déjà offert.

Perspectives

Dans cette thèse, nous avons déjà commencé à travailler avec une nouvelle approche de résolution qui consiste à créer des groupes virtuels de blessés avant de commencer le processus d'évacuation.
Pour les travaux futurs, nous pensons qu'il est nécessaire d'apporter d'autres améliorations en :

1. Développant une approche méta-heuristique pour résoudre les groupes multi-prioritaires.
2. Ajoutant une capacité hospitalière dynamique.
3. Mettant en œuvre la notion d'urgence des blessés et de priorité de transport.

Bibliography

Sbihi A., Eglese R. W. (2007). *The Relationship between Vehicle Routing & Scheduling and Green Logistics - A Literature Survey.*

Billington L. H. (1993). Material management in decentralized supply chains. *Operations Research, 41*, 835-847.

Blum C. (2010). Hybrid metaheuristics. *Computers & Operations Research, 37*(3), 430-431.

Byung Hoon Oh, Kwangyeon Kim, Han-Lim Choi, Inseok Hwang (2017). Integer Linear Program Approach for Evacuation of Disaster Victims with Different Urgency Levels, IFAC. *Federation of Automatic Control, 50*(1), 15018-15023.

Chawis Boonmee, M. A. (2017). Facility location optimization model for emergency humanitarian logistics. *International Journal of Disaster Risk Reduction, 24*, 487-498.

Christofides, N. M. (1979). The vehicle routing problem. In N. A. Christofides, *Combinatorial Optimization* (pp. 315-338). Chichester: Wiley.

Christofides, N. M. (1981). Exact algorithms for the vehicle routing problem based on the spanning tree and shortest path relaxations. *Mathematical Programming, 20*, 255-282.

Christofides, N. M. (1981). State-space relaxation procedures for the computation of bounds to routing problems. *Networks, 11*, 145-164.

Christofides, N. M. (1989). Vehicle routing: Practical and algorithmic aspects. In C. (. Van Rijn, *Logistics: Where Ends Have to Meet* (pp. 30-48). Oxford: Pergamon.

Christopher, M. (1992). *Logistics and supply chain managment.* London: Pitmans.

Christopher, M. (1996). From brand values to customer value. *Journal of Marketing Practice: Applied Marketing Science, 2*(1), 55-66.

Colin, M. &. (1981). *La logistique au service de l'entreprise.*

Croom, S. R. (2000). chain management: An Analytical framework for critical literature review. *European Journal of Purchasing & Supply Management, 6*, 67-83.

D.Eksioglu, F. S. (2010). Optimizing the use of public transit system during no-notice evacuation of urban areas. *Computers & Industrial Engineering, 57*(4), 488-495.

Dantzig, G. (1987). *Origins of the simplex method.* Stanford: Stanford University.

Devadas, S., & Newton, A. (1989). Algorithms for hardware allocation in data path synthesis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 8*, 768-81.

Doig, A. H. (1960). An automatic method of solving discrete programming problems. *Econometrica, 28*(3), 497-520.

Dorigo, M. (1992). *Optimization, learning and natural algorithms.* Milano: PhD thesis, Dipartimento di Elettronica, Politecnico di Milano.

Dorigo, M. a. (1997). Ant colony system: A cooperative learning approach to traveling salesman problem. *IEEE Transactions on Evolutionary Computation, 1*(1), 56-66.

Ejday, M. (2011). *Optimisation Multi-Objectifs à base de Métamodèle pour les procédés de mise en forme. Thèse de doctorat.* Paris: École des MINES ParisTech.

El Sayed AR, E. C. (2018). Efficient 3D point clouds classification for face detection using linear programming and data mining. *The Imaging Science Journal, 66*(1), 23-37.

Feo, T. &. (1995). Greedy randomized adaptive search procedures. *Journal of global optimization, 6*(2), 109-133.

Glover, F. (1989). Tabu search–Part I. *ORSA Journal on Computing 1*, 190-206.

Glover, F. (1990). Tabu search–Part II. *ORSA Journal on Computing 2*, 4-32.

Herrera, C. (2011). *Cadre générique de planification logistique dans un contexte de décisions centralisées et distribuées, Ph.D. thesis.* Nancy I: Université Henri Poincaré.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems.* Cambridge: MA: MIT Press. Second edition .

Huang, G. L. (2003). The impacts of sharing production information on supply chain dynamics: a review of the literature. *International Journal of Production Research, vol 41, No. 7*, 1483-1517.

Huang, G. L. (2003). The impacts of sharing production information on supply chain dynamics: a review of the literature. *International Journal of Production Research, 41*, 1483-1517.

J. Thomas, P. G. (1996). Coordinated supply chain management. *European Journal of Operational Research, 94*, 1-15.

João Coutinho-Rodrigues, L. T.-A. (2012). Solving a location-routing problem with a multiobjective approach. *Engeneering Science and Technology, an International Journal: the design of urban evacuation plans, 22*, 206-218.

Juan A. Díaz, D. E.-F.-V.-S.-R. (2017). GRASP and hybrid GRASP-Tabu heuristics to solve a maximal covering location problem with customer preference ordering. *Expert Systems with Applications, 82*, 67-76.

Keliang Chang, H. Z. (2017). Multiobjective Location Routing Problem considering Uncertain Data after Disasters. *Discrete Dynamics in Nature and Society. Volume 2017, Article ID 1703608*.

Laguna, M. &.-V. (1991). search heuristic for just-in– time scheduling in parallel machines. *Journal of Intelligent Manufacturing, vol 2, issue 4,* , 253-260.

Lozano, M. &.-M. (2010). Hybrid metaheuristics with evolutionary algorithms specializing in intensification and diversification: Overview and progress report. *Computers & Operations Research, volume 37, issue 3, 37*(3), 481-497.

M. Chen, L. C.-H. (2007). Traffic signal timing for urban evacuation. *Journal of Urban planning and development, 1333*(1), 30-42.

M. Christopher, L. M. (1998). Educational development for marketing logistics. *International Journal of Physical Distribution and Logistics Management, 28*(4), 234-241.

Maria. (1997). *Introduction to Modeling and Simulation, In Proceedings of the Winter Simulation Conference.*

Mori, A. K. (2015). Swarm Optimization / Greedy-Search Algorithm for Helicopter Mission Assignment in Disaster Relief. *Journal of Aerospace Information Systems, 12*(10), 646-660.

Noraimi Azlin Mohd Nordin, Z. A. (2012). Finding Shortest Path of the Ambulance Routing: Interface of A* Algorithm using C# Programming. *IEEE Symposium on Humanities, Science and Engineering Research*, 1569-1573.

Peck, M. C. (1-13). BUILDING THE RESILIENT SUPPLY CHAIN. *International Journal of Logistics Management, 15*(2), 2004.

R. R. Lummus, R. J. (1998). Strategic supply chain planning. *Production and Inventory Management Journal, vol 39 (3)*, 49-58.

R. R. Lummus, R. J. (1998). Strategic supply chain planning. *Production and Inventory Management Journal, 39*(3), 49-58.

Raidl, J. P. (2005). Combining metaheuristics and exact algorithms in combinatorial optimization: A survey and classification. *Proceedings of the First International Work-Conference on the Interplay Between Natural and Artificial Computation. Volume 3562 of Lecture Notes in Computer Science*, 41-53.

Ramser, G. B. (1959). The Truck Dispatching Problem. *Management Science, vol. 6, No. 1, 6*(1), 80-91.

Silva, C. S. (2006). Distributed Optimization of a logistic system and its suppliers using ant colonies. *International Journal of Systems Science, vol 37(8), 37*(8), 503-512.

Stein M, H. A. (1999). Medical consequences of terrorism. The conventional weapon threats. *Surg Clin North Am, 79*, 1537-1552.

Talbi, E.-G. (2002). A Taxonomy of Hybrid Metaheuristics. *Journal of Heuristics, volume 8, issue 5, 8*(5), 541-564.

Tan, K. (2001). A framework of supply chain management literature. *Journal of Purchasing & Supply Management, 7*, 39-48.

Wei Yi, L. O. (2007). A dynamic logistics coordination model for evacuation and support in disaster response activities. *European Journal of Operational Research, 179*, 1177-1193.

Wex, F. S. (2014). Emergency response in natural disaster management: Allocation and scheduling of rescue units European. *Journal of Operational Research, 255*(3), 697-708.

Xin Ma, Y. S. (2010). Min-max Robust Optimization for the Wounded Transfer Problem in Large-scale Emergencies. *IEEE, Chinese Control and Decision Conference (CCDC)*.

Yadollahnejad, V. (2017). Allocation and Vehicle Routing for Evacuation Operations: A Model and a GRASPHeuristic. *Journal of Urban Planning and Development, 143*(4).

Zabinsky, H. O. (2010). Stochastic optimization of medical supply location and distribution in disaster management. *International Journal of Production Economics, 126*(1), 76-84.

Zhang, X. (2017). Study on rapid evacuation in high-rise buildings. *Engineering Science and Technology, an International Journal, 20*(3), 1203-1210.

[i] https://fr.wikipedia.org/wiki/Algorithme_de_colonies_de_fourmis#/media/File:Aco_branches.svg
[ii] http://en.wikipedia.org/wiki/Hill_climbing
[iii] https://www.livescience.com/39110-japan-2011-earthquake-tsunami-facts.html
[iv] http://edition.cnn.com/2013/12/12/world/haiti-earthquake-fast-facts/index.html
[v] https://www.livescience.com/22522-hurricane-katrina-facts.html
[vi] http://www.bbc.com/news/world-middle-east-1110773947
[vii] https://www.theguardian.com/world/2016/feb/11/report-on-syria-conflict-finds-115-of-population-killed-or-injured
[viii] https://www.sciencedirect.com/science/article/pii/S2211419X15000348
[ix] http://www.lefigaro.fr/actualite-france/2015/11/14/01016-20151114ARTFIG00252-six-attaques-en-33-minutes-chronologie-d-un-massacre.php
[x] https://www.britannica.com/event/Indian-Ocean-tsunami-of-2004
[xi] https://www.britannica.com/event/Sichuan-earthquake-of-2008
[xii] http://www.hko.gov.hk/m/article_e.htm?title=ele_00049
[xiii] https://www.livescience.com/39110-japan-2011-earthquake-tsunami-facts.html
[xiv] https://www.theguardian.com/commentisfree/2013/jun/28/india-floods-man-made-disaster
[xv] https://ideas.repec.org/a/spr/nathaz/v89y2017i1d10.1007_s11069-017-2974-4.html
[xvi] https://reliefweb.int/map/china/flooding-china-28-july-2016
[xvii] https://knowledge.aidr.org.au/resources/bushfire-black-saturday-victoria-2009/
[xviii] https://www.ncbi.nlm.nih.gov/pubmed/15187837
[xix] http://www.lorientlejour.com/article/829379/la-terreur-frappe-deux-mosquees-sunnites-a-tripoli-lors-de-la-priere-du-vendredi.html
[xx] http://edition.cnn.com/2017/11/24/africa/egypt-sinai-mosque-attack/index.html