TABLE DES MATIÈRES

LISTE DES TABLEAUX

Page

LISTE DES FIGURES

Page

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

BAP: Berth allocation problem

CAP: Crane assignment problem

BACAP: Berth allocation Problem and Crane assignment problem.

MIP: Mixed Integer Programming.

EGD: Extended Great Deluge.

ABC: Artificial Bee Colony.

SA: Simulated annealing.

INTRODUCTION

Le transport maritime joue aujourd'hui un rôle prédominant dans les transports internationaux. Au moins 90% du volume total de marchandises transportées empruntent la voix maritime. Ce taux varie selon les régions. Les 10% qui n'empruntent pas les modes de transport maritime se situent essentiellement en Europe Occidentale et en Amérique du Nord. Dans ces pays, les principaux partenaires commerciaux sont directement reliés entre eux par un réseau de transport intermodal bien développé.

Le transport maritime, et en conséquence les services portuaires, sont donc vitaux pour les économies des pays en développement. En effet le commerce international est l'un des principaux moyens de moderniser un pays et le prix du transport est devenu un facteur déterminant de la compétitivité de l'Economie des Nations.

D'un autre coté, la fiabilité et la qualité des services à terre sont des facteurs décisifs du choix des armateurs (les exploitants des navires pour la navigation commerciale) d'un port par rapport à d'autres, d'où la nécessité d'une bonne gestion des terminaux portuaires pour concurrencer les autres pays dans cette course de compétitivité.

Un terminal portuaire est constitué d'un poste à quai, ou d'un groupe de postes à quai, permettant le stationnement et l'opération des navires affectés à un trafic particulier, et complété par les installations terrestres nécessaires à l'exploitation de ce trafic. On trouve une bonne illustration de cette notion de terminal pour le trafic *conteneurisé*: il y a maintenant dans presque tous les ports un ou plusieurs terminaux à conteneurs, ce qui correspond à l'adaptation du port à la tendance à l'accroissement de la '*conteneurisation'*.

Bien que de nos jours il existe encore, notamment sur les lignes desservant des pays en développement, de nombreux navires classiques transportant des marchandises diverses, on peut estimer qu'il n'y aura dans l'avenir que trois grandes catégories de navires de marchandises: les vraquiers (solide /liquide), les porte-conteneurs et les navires spécialisés classiques ou rouliers (colis lourds, encombrants, produits spécifiques, voitures, néo-vrac autrement dit les marchandises non *conteneurisables*).

Les terminaux à conteneurs sont donc les terminaux dédiés aux navires porte-conteneurs, et sont les aires où les conteneurs sont transportés d'un point à un autre en utilisant différents équipements de manutention. L'importance de ces terminaux ne cesse de s'accroître au même niveau que le défi des nouvelles technologies du transport maritime à construire de plus grands navires. Dans ce contexte de développement soutenu du trafic conteneurisé, l'exploitation de terminaux à conteneurs est devenue une activité de premier plan. Et afin de concurrencer dans cet environnement, le terminal à conteneurs doit être géré efficacement.

Dans notre cadre, une bonne gestion du terminal revient à minimiser le temps passé par un conteneur ou un navire dans le port. Plus précisément, la gestion fine de manutention des conteneurs pour chargement/déchargement apparaît comme un problème à part entière qui doit être étudié en profondeur. Cette manutention est le premier maillon de la chaine en import et le dernier maillon en export.

La logistique dans un terminal à conteneurs

Les processus généraux dans un terminal à conteneurs peuvent être décrits comme une séquence d'événements (Fig.0.1) à partir de l'arrivée d'un navire chargé de conteneurs à l'import jusqu'au départ des conteneurs vers les clients ou selon la chaîne inverse, c-à-d à partir de l'arrivée des conteneurs destinés à l'export jusqu'à leur départ à bord du navire.

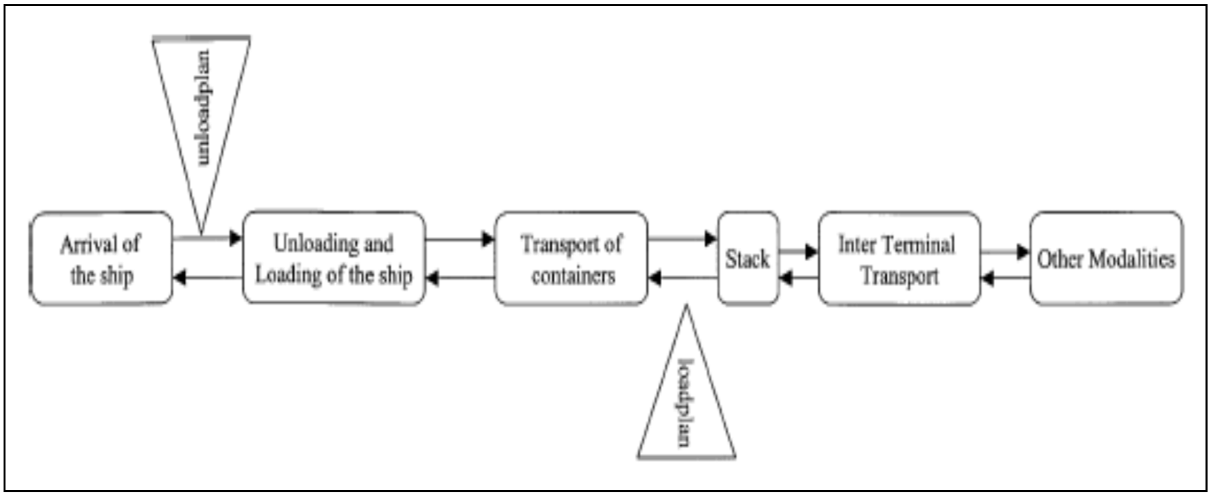

Figure 0.1 – Processus général dans un terminal à conteneurs (Vis and Koster 2003)

Quand un navire arrive au port, il doit tout d'abord trouver un emplacement pour accoster (Berth), les conteneurs à bord doivent être déchargés par des grues de quais (Quay Cranes). Ces grues prennent les conteneurs du navire et les déposent sur la plate-forme, ensuite, ils sont transportés jusqu'aux piles (Stacks) dans les zones de stockage. Cette zone est servie par les grues de cour (Gantry cranes) ou les chariots-cavaliers (straddle carrier). Après une certaine période de temps, les conteneurs sont retirés des piles et sont transportés par véhicules à d’autres types de transport (Trains, camions, autres navires).

La figure 0.2 résume ces différentes opérations et présente également les alternatives par les différents engins de manutention du terminal.

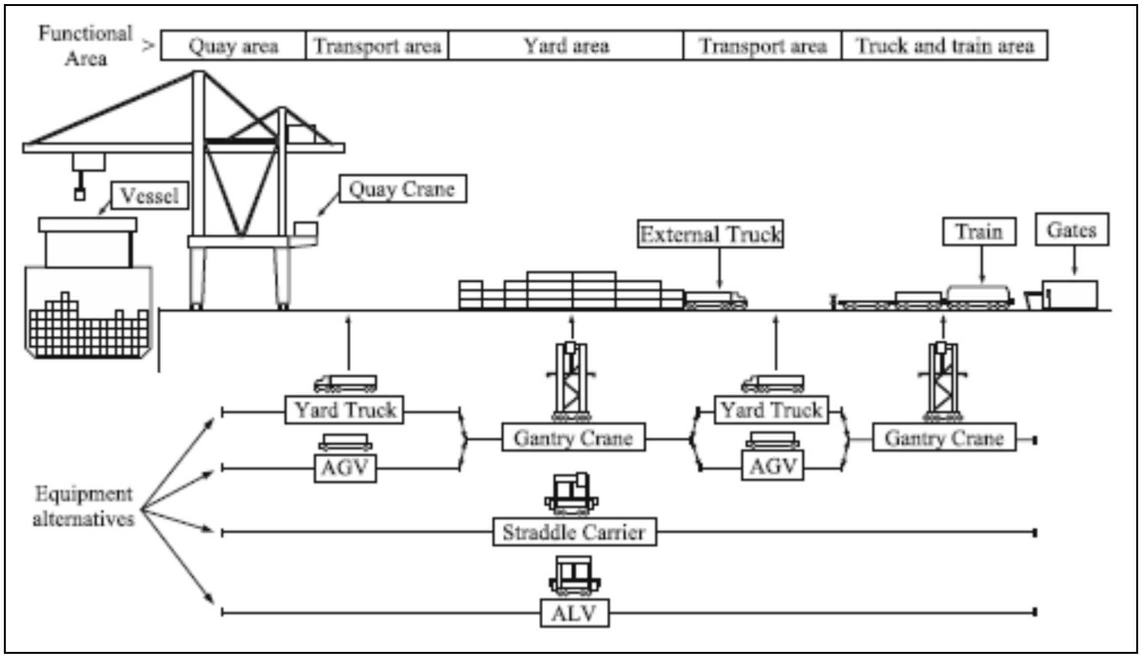

Figure 0.2 – Opérations et équipements dans un terminal à conteneurs (Meisel.2009)

Les opérations, dans un terminal à conteneurs, deviennent de plus en plus compliquées, en raison de la complexité et du nombre d'entités impliquées dans son fonctionnement. De ce fait, les gestionnaires de terminaux sont confrontés de plus en plus à plusieurs prises de décisions importantes pour que le terminal puisse être compétitif.

Parmi les décisions les plus critiques et qui sont prises au début de la chaîne, nous retrouvons les problèmes tactiques de planification et précisément le problème intégré d'allocation de zones d'accostage (Berth Allocation) et d'assignation de grues de chargement (Crane Assignment). C'est dans ce cadre que s'inscrit ce travail de recherche, et c'est précisément la modélisation mathématique puis la résolution de ce problème par les méthodes approchées d'optimisation qui va nous intéresser tout au long de ce travail.

Après avoir découvert le monde des opérations portuaires à travers la lecture d'articles scientifiques, nous nous sommes fixés comme objectif d'aller sur le terrain pour découvrir réellement ce qui se passe. L'occasion s'est donc présentée en l'automne 2015 quand nous avons passé un stage de deux mois au terminal à conteneurs de Rades en Tunisie.

La réalité était plus complexe que ce que les modèles mathématiques, trouvés dans la littérature, montraient. Les problèmes étaient réellement intimement liés, et le contexte tunisien rendait les choses plus difficiles. Finalement, le cas du terminal portuaire de Radès présentait une étude très intéressante pour notre cadre de recherche.

Les contributions scientifiques de ce travail de recherche ont été un article accepté et présenté lors d'une conférence internationale avec comité de lecture et trois articles de journaux dont un présentait une nouvelle méta-heuristique pour la résolution d'un modèle non linéaire pris comme référence dans ces deux variantes mono et multi-objectif. Les deux autres articles traitent et modélisent le problème intégré de la double allocation de ressources dans le cas du terminal tunisien de Rades avec deux variantes de l'assignation des grues. Un est accepté et l'autre est soumis pour publication.

Nous commençons par une revue de littérature qui survole les problèmes de planification dans un terminal à conteneurs et les décisions rencontrées dans ce contexte et découlant de l'activité logistique des conteneurs sur un terminal. Nous mettrons l'accent ensuite sur les problèmes intégrés de *Berth Allocation and Crane Assignment* en faisant un tour d'horizon des particularités des modèles proposés et des méthodes et approches de résolution qui leurs ont été appliquées.

REVUE DE LITTÉRATURE ET STRUCTURE DE LA THÈSE

Comme a été décrit pour introduire ce travail, le processus des opérations dans un terminal portuaire est une suite d'événements à partir de l'arrivée du navire jusqu'à son départ. Il s'avère donc difficile de considérer les décisions dans un terminal, indépendantes les unes des autres.

Des recherches ont, à cet effet, mis l'emphase sur la gestion intégrée des problèmes rencontrées dans un terminal portuaire. La majorité des études sur le sujet portent sur la conception de systèmes d'aide à la décision et sur la simulation. Plusieurs revues de littérature ont présenté les différents travaux effectués dans ce domaine, en décrivant les décisions à prendre dans un terminal portuaire et en les classant en trois niveaux de prise de décision, à savoir le niveau stratégique, tactique et opérationnel. La classification la plus répandue dans la plupart des travaux de recherche publiés étant une classification basée sur l'ordre séquentiel des opérations, c'est-à-dire, opérations relatives à l'arrivée des navires, opérations de chargement/déchargement, opérations de transfert des conteneurs entre les navires et les zone de stockage et finalement les opérations de stockage.

Tel que décrit par Henesy dans la figure 0.3, dès l'arrivée du navire, le gestionnaire est confronté au problème de l'allocation de la zone d'accostage dit *Berth allocation Problem* (BAP). Les opérations de chargement/déchargement nécessitent une prise de décision d'un niveau plus opérationnel pour l'assignation des grues de quai (Crane Assignemnt Problem CAP). Les zones de stockage doivent aussi être bien gérées et bien desservies par les engins de manutention dédiés tel que les cavaliers gerbeurs (starddle carrier)

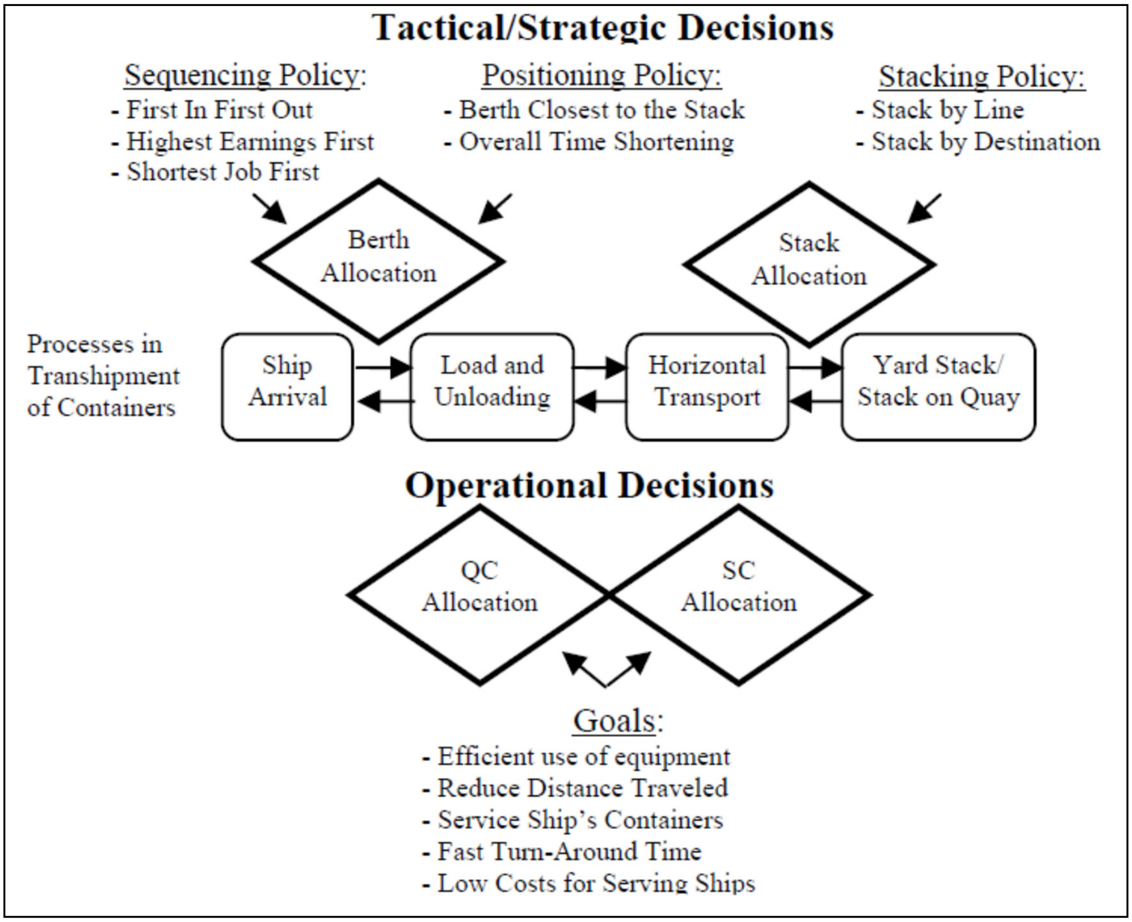

Figure 0.3 – Types de décisions dans un Terminal à conteneurs (Henesy, 2006)

Un autre type de classification des problèmes dans un terminal, présentée par Meisel (2009) sur la figure 0.4 est celle qui considère le terminal comme 3 zones inter-reliées, qui sont la zone de bord (*Seaside*), la cour (*Yard*) et la zone d'opérations terrestres (*landside*). Dans chacune de ces zones, plusieurs prises de décision sont à considérer.

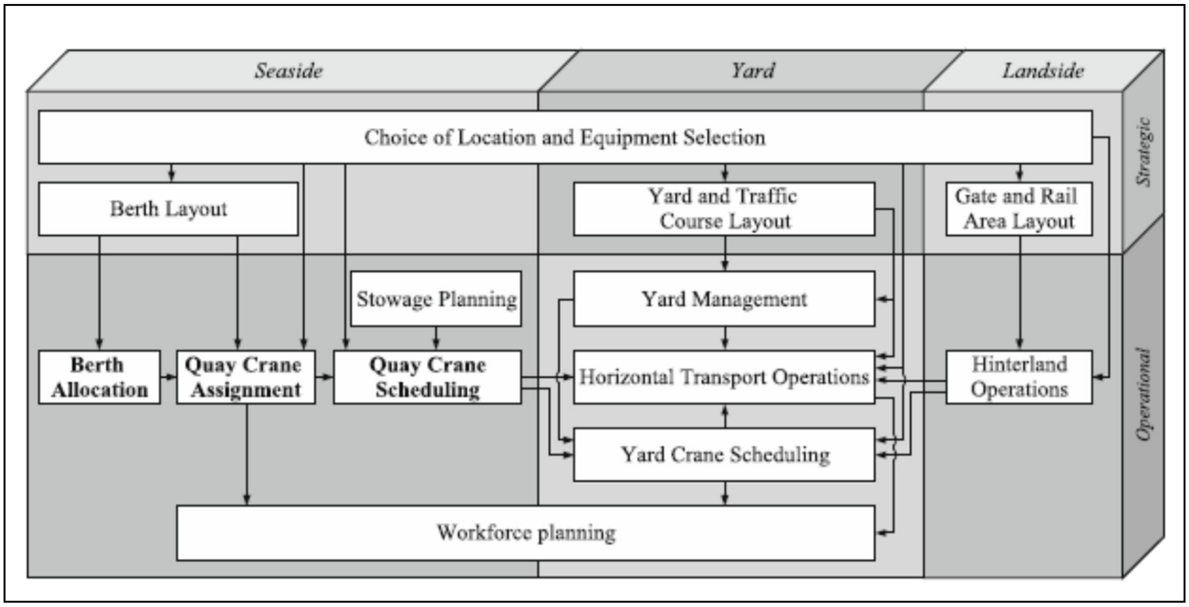

Figure 0.4 – Problèmes de planification dans un terminal à conteneurs (Meisel, 2009)

Parmi les auteurs qui ont commencé le recensement problèmes des terminaux conteneurs, il y a Vis et Kostner (2003) qui ont décrit les opérations sur un terminal de transbordement avec tous les équipements de manutention requis et les prises de décisions qui lui sont relatives pour leur gestion. Steenken et al. (2004) ont présenté une autre classification des opérations dans un terminal allant de la planification de l'arrivée du navire, passant par les problèmes de stockage jusqu'aux problèmes d'optimisation de transport de véhicules.

Rachidi et al. (2006) ont présenté des tableaux classifiant les auteurs et les problèmes de prise de décision dans un terminal avec leurs caractéristiques et la méthode de résolution adoptées. Stahlbock et al. (2008), ont repris l'état de l'art de Steenken (2004) et l'ont mis à jour, en gardant la même classification des problèmes.

Murty et al. (2005) décrivent les différentes sections d'un terminal avec les équipements en présentant quelques indicateurs de performance pour les terminaux portuaires. Ils proposent aussi quelques stratégies utilisées pour concevoir un système d'aide à la décision au port de Hong Kong.

Huang et al. (2014) ont mis l'accent sur les problèmes d'allocation de ressources dans un terminal, y compris le problème d'une seule ressource à la fois qui est l'allocation des zones d'accostage (Berth allocation Problem) et l'allocation de plusieurs ressources simultanément comme le problème intégré BAP-CAP.

Salido et al. (2011) focalisent essentiellement sur 3 types de problèmes qu'ils supposent inter-reliés (illustrés par la figure 0.5) à savoir le BAP le CAP et le problème de stockage de conteneurs dans la zone dédiée (*Container stacking problem*). Ils présentent un système d'aide à la décision qui se base sur des heuristiques et la métaheuristique ***GRASP*** *(Greedy randomized adaptive search procedure)* pour la proposition de solutions.

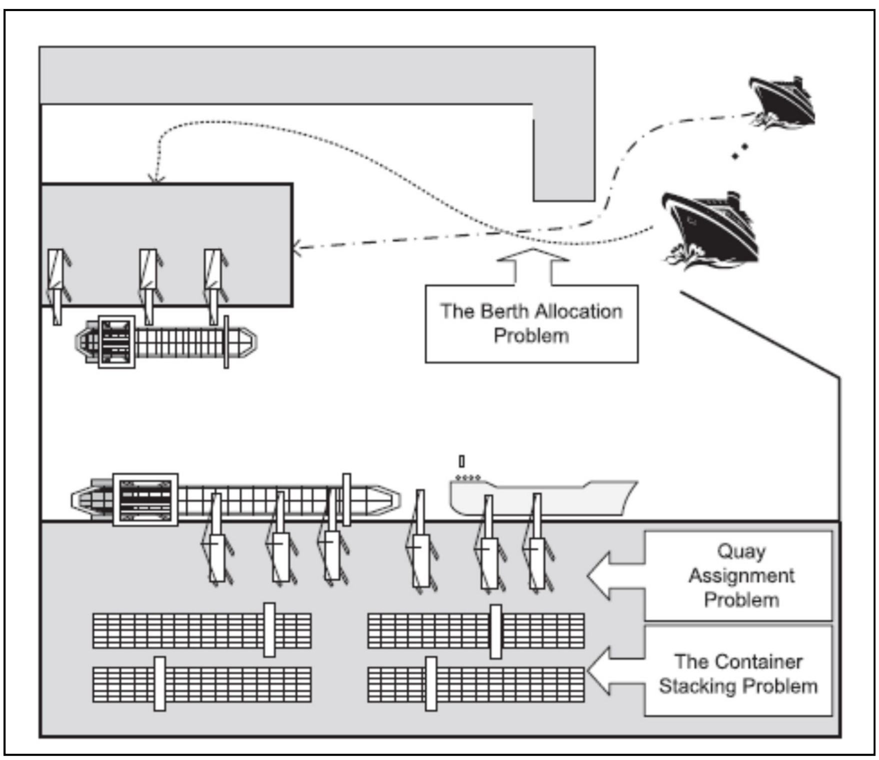

Figure 0.5 – Schématisation du problème intégré BAP

Les approches basées sur la simulation ont été l'alternative pour contourner la difficulté d'utiliser des méthodes analytiques pour modéliser et résoudre les problèmes intégrés. En effet, les recherches dans cet axe prennent de plus en plus d'ampleur et ont comme finalité de simuler les différentes stratégies de prise de décision dans un port. Selon Hassan (1993), un modèle de simulation peut être utilisé pour déterminer l'effet des variations des options opérationnelles, technologiques ou d'investissement sur les indicateurs d'un port. En 1987, Asim et al., ont pu confirmer que la simulation est un outil intéressant pour les décisions portuaires. Dans leur travail, l'emphase a été mise sur le niveau stratégique à savoir estimer le nombre adéquat de zones d'accostage ainsi que les ressources de manutention nécessaires pour un port. Razman et al. (2000) ont travaillé sur le port de Kelang en Malaisie, pour simuler les opérations d'allocation des zones d'accostage et des 2 types de ressources qui sont les grues et les premiers vehicules de transfert (*prime movers*). L'outil de simulation est le logiciel ARENA.

Arango et al. (2011), simulent les opérations dans le port de Séville en Espagne. Ils proposent une application de BAP (modélisé mathématiquement) qui intègre une résolution par les algorithmes génétiques avec une simulation par Aréna.
Chang, et al. (2008), utilisent la même approche, soit l'intégration d'un module d'optimisation basé sur des heuristiques avec une procédure de simulation utilisant le logiciel Simtalk. Encore une fois, un exemple réel a été traité en utilisant le terminal de Shangai.

Une récente publication, Ji et al. (2015) traite le problème d'allocation des zones d'accostage et des grues par la simulation. Ils étudient les différentes stratégies d'allocation et leur impact sur les 3 indicateurs de performance *BUR : Berth Utilization Rate, VOR : Vessel Operation Rate et CUR* : *Crane Utilisation Rate* en utilisant la simulation par Monte-Carlo.

Berth Allocation Problem (BAP)

Définition du problème

Quand un navire arrive au port, il attend son tour pour accoster au quai. Les sections du quai réservées où l'accostage s'effectue sont appelées zones d'accostage. Les problèmes d'allocation de ces zones consistent à assigner, d'une manière optimale, les navires entrant aux ports à ces postes d'accostage. Le responsable logistique est confronté alors à deux décisions : ***Où*** et ***Quand*** le navire doit-il accoster?

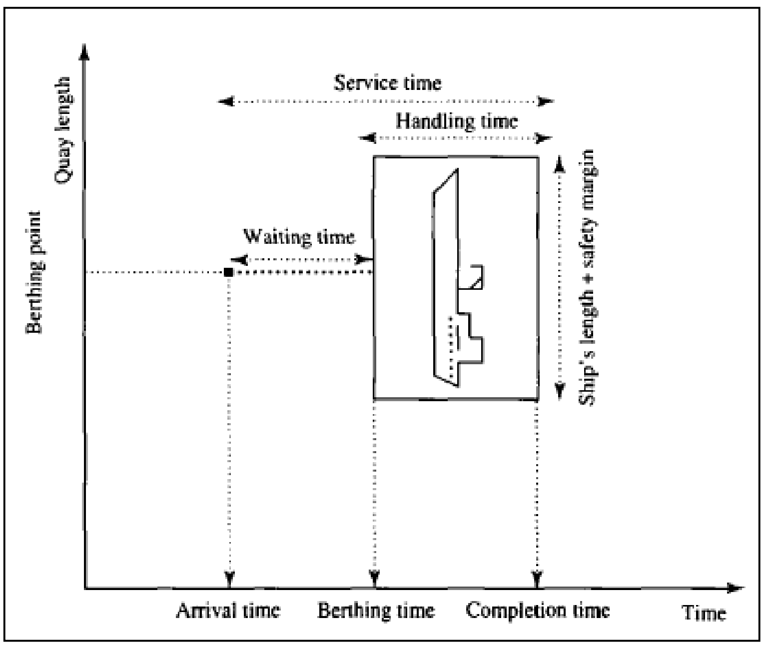

Figure 0.6 – Diagramme bidimensionnel Temps-Espace pour le BAP

Classiquement, le problème est représenté par un diagramme bidimensionnel temps-espace (figure.6) où les navires sont des rectangles dont les dimensions sont leurs longueurs respectives (y compris les marges de sécurité) et le temps de manutention correspondant au déchargement de leurs cargaisons. Ces rectangles doivent être placés dans l'espace de décision sans chevauchement et satisfaisant certaines contraintes. Pour la dimension spatiale, il y'a par exemple la contrainte relative à la profondeur de l'eau. Pour la dimension temporelle, les contraintes sont exprimées comme des fenêtres de temps relatives au temps de service (temps de chargement/déchargement). Ce temps de manutention dépend de la position du point d'accostage et est fonction de la distance séparant cette zone d'accostage et la zone d'entreposage des conteneurs qui vont être chargés ou déchargés. Cette dépendance affecte fortement la performance des services du port.

L'horizon de planification pour ces problèmes est d'une semaine mais le plan d'accostage doit être mis à jour quotidiennement en raison des événements imprévus qui peuvent survenir (maintenance de certaines zones du quai, non concordance des arrivées effectives avec les planifications). Le temps d'arrivée des navires est estimé à l'avance, et chaque navire a sa propre fenêtre de temps déterminée par son temps d'arrivée et la durée maximale allouée à son service (chargement/déchargement).

Les responsables de logistique veulent alors optimiser à la fois les coûts du port et des clients (armateurs) qui sont liés au temps de service. L'objectif ultime des problèmes d'allocation des zones d'accostage est donc d'optimiser l'efficacité du service sur le port pour tous les navires.

Revue de littérature relative au BAP

Il y a presque unanimité sur le fait que l'allocation des zones d'accostage a un impact primordial sur l'efficacité des opérations sur le terminal puisque, ces zones d'accostage sont les ressources les plus importantes dans un terminal à conteneurs. De ce fait une allocation efficace des zones aux navires entrants améliore la satisfaction des armateurs et accroît la productivité du terminal. Les travaux publiés qui traitent du problème d'allocation des zones d'accostage ont démarré avec « *the efficient Planning of Berth Allocation for container terminal in Asia* » (Imai et al. 1997). Les auteurs ont voulu mettre l'accent sur le niveau opérationnel de la prise de décision dans la gestion d'un terminal.

Se basant sur le rapport annuel du port de Singapore de 1994, indiquant que l'un des problèmes majeurs de planification était de décider comment accoster les navires arrivant au port aux différentes sections du quai en respectant certaines contraintes, Lim (1998) présente une description du *Berth Planning Problem* (BPP) ainsi qu'une représentation géométrique (digramme bidimensionnel espace-temps) et une représentation de graphes du problème. Il s'appuie sur les travaux de (Garey et Johnon, 1979) pour conclure que le problème est NP complet et que la résolution par une méthode approchée (heuristique) est plus adaptée.

Dans la littérature, le problème BAP a été étudié selon deux variantes : statique et dynamique. Le problème est dit statique si on suppose que les navires arrivent au port avant que les sections de quais (berths) ne deviennent disponibles. La variante dynamique, la plus réaliste, permet que les navires arrivent au fur et à mesure de la planification, càd, avant ou après la disponibilité des zones d'accostage (Hansen, 2008). Les deux versions du problème BAP ont été étudiées chacune majoritairement selon deux classes de modèles : discret et continu. En effet, les zones d'accostage sont considérées soit comme ressources discrètes, soit comme ressources continues, dépendamment de la construction et la topologie du terminal. Selon le scénario d'une ressource discrète, les zones d'accostages sont des ressources individuelles et chaque navire doit être assigné à une seule et unique section ou zone d'accostage. Selon le scénario d'une ressource continue, le quai, lui-même, est traité comme une seule grande section à laquelle plusieurs navires peuvent s'accoster simultanément. Comparés aux modèles discrets, les modèles continus sont plus flexibles dans le sens où les responsables logistiques sont plus libres de décider quant à l'assignation des zones d'accostage. Il y a aussi ce que Bierwith et Meisel (2010) appellent le modèle hybride.

La figure 0.7 tirée de leur travail (Bierwirth and Meisel 2010) présente les différentes configurations possibles de répartition des quais. Le modèle discret pourrait être celui de (a) ou (b), le quai continu est présenté par (c), la configuration hybride est soit dans le cas où un grand navire accoste sur deux sections du quai (d), soit que deux petits navires trouvent place sur une même section (e). Il y'a aussi les postes à quais dédiés (f).

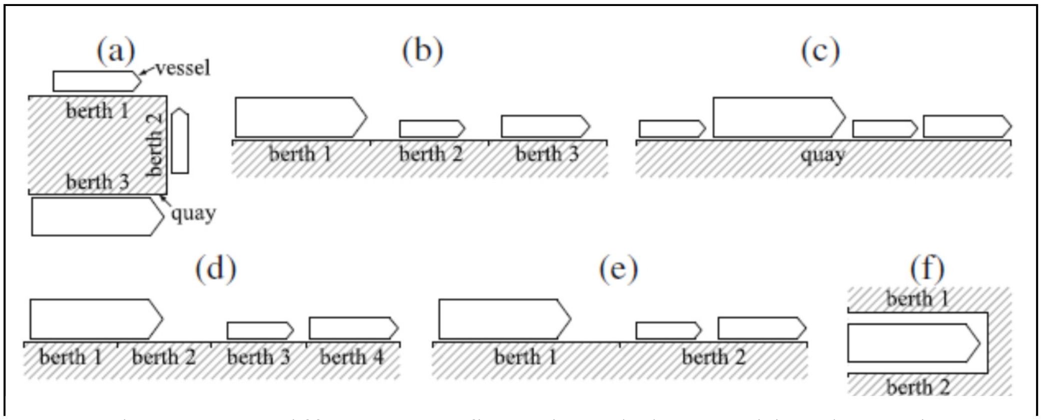

Figure 0.7 – Différentes configurations de la répartition des quais

Pour faire le parallèle avec l'ordonnancement d'ateliers, plusieurs auteurs considèrent que le problème discret d'allocation des zones d'accostage peut être modélisé comme un problème d'ordonnancent de machines parallèles, où les navires sont considérés comme des tâches et les zones d'accostage comme des machines. Le problème continu quant à lui, est considéré comme un problème de découpe/placement (*Cutting-Stock Problem*) bidimensionnel avec des contraintes additionnelles. Dans les deux cas, le problème d'allocation des zones d'accostage est un problème NP-complexe (Garey et Johnson, 1979).

Comme introduit plus haut, étant donné un nombre de navires arrivant à un port dans un horizon de temps de planification, le problème d'allocation des zones d'accostage est un problème de détermination du temps et de l'espace de chacun des navires, en satisfaisant un nombre de contraintes spatiales et temporelles, afin d'optimiser certaines opérations. Dans la littérature, plusieurs objectifs pour cette optimisation ont été considérés.

Imai et al. (2001, 2003, 2005, 2007) minimisent le temps total de service des navires. Ce temps de service pour chaque navire inclut le temps d'attente entre l'arrivée au port et l'accostage, le temps de chargement et/ou déchargement des conteneurs. Guan et al. (2002) développent un modèle pour minimiser la somme des durées de service pondérées (*weighted completion time*) des navires accostés. Hansen et al. (2007) élaborent un modèle pour minimiser le coût total d'accostage engendré par l'attente, le chargement/déchargement et la pénalité du non-respect du planning.

Kim et Moon (2003) proposent un modèle linéaire a variables mixtes pour minimiser le coût de pénalité résultant des retards pour les départs des navires et des manutentions additionnelles dues à une localisation non-optimale des navires car selon leur modèle, chaque navire possède une zone d'accostage optimale. Park et Kim. (2003) modélisent le problème de la même manière que Kim et and Moon (2003) en considérant que les coûts additionnels sont dus au début au plutôt ou au plus tard des opérations de chargement/déchargement par rapport à l'estimation du temps d'arrivée. Li et al. (1998) minimisent le *Makespan* de la planification. Imai et al. (2003) abordent le problème d'allocation des zones d'accostage en considérant la priorité de certains navires par rapport aux autres. Lim (1998) présente une approche différente du problème en minimisant l'espace maximum réservé pour l'accostage des navires.

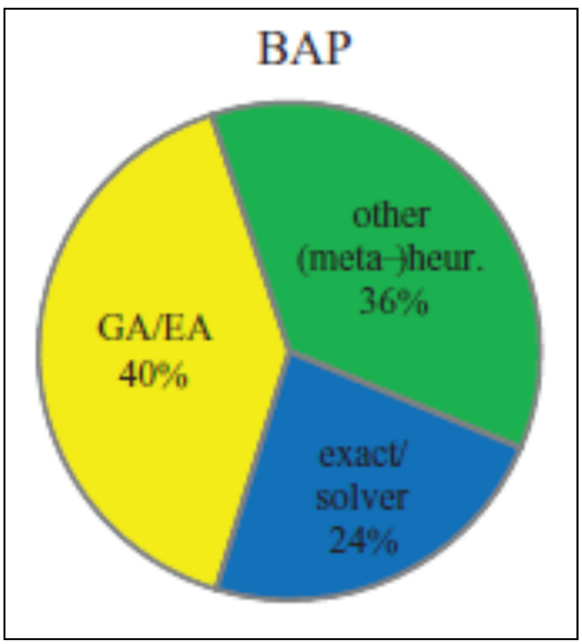

Figure 0.8 – Répartition des méthodes
de résolution du BAP dans la littérature
(Bierwirth and Meisel. 2015)

Les modélisations mathématiques et la résolution des problèmes d'allocation des zones d'accostage ont été diversifiées dans la littérature. Des méthodes exactes ainsi que d'autres approchées (heuristiques et méta-heuristiques) ont été appliquées. Selon Bierwirth at Meisel (2015), et comme présenté par la figure 8, seulement le quart des approches utilisent les solveurs et les méthodes exactes. Le reste des méthodes de résolutions se répartissent entre heuristiques et méta-heuristiques avec une grande part (40 %) aux algorithmes évolutionnaires comme les algorithmes génétiques.

Nishimura et al. (2001) ont par exemple présenté une modélisation par programmation non-linéaire à variables entières et une résolution par algorithme génétique pour la variante dynamique discrète du problème d'allocation en minimisant le temps total de service. Une extension de ce même problème a été proposée par Imai and al. (2003) pour considérer la priorité de service. Le modèle est non linéaire et les auteurs ont proposé une relaxation lagrangienne pour rendre le problème celui d'une assignation quadratique et pour le résoudre, ils ont opté pour l'algorithme génétique puisque les méthodes exactes ne résolvent pas ce genre de problèmes efficacement. Les simulations des résultats ont été effectuées sur des horizons de planification différents (de 1 à 6 jours).

Kim et Moon (2003) ont adopté une formulation par programmation linéaire mixte pour le problème d'allocation de zone d'accostage dans le cas continu. La résolution s'est faite par un logiciel commercial (Lindo) pour un problème de petite taille et la méta-heuristique du recuit simulé a été proposée pour la résolution des problèmes réels de plus grande taille.

Hansen et al. (2007) ont modélisé le problème discret-dynamique par programmation linéaire mixte et l'ont résolu par une recherche de voisinage variable. Ils ont comparé leurs résultats à ceux obtenus par l'heuristique de recherche locale *Multi start*, par la méta-heuristique de l'algorithme génétique et par l'algorithme de *Memetic Search*. Ils ont conclu que les résultats obtenus par une recherche de voisinage variable sont les plus performants. Cordeau et al. (2005) ont adapté le modèle du BAP dynamique d'Imai et al. (2001) et ont développé une heuristique basée sur la recherche taboue pour la résolution du problème dans le cas discret puis l'ont étendu pour le cas continu.

Wang et al. (2007) ont transformé le problème d'allocation des zones d'accostage en un problème de prise de décision multi étapes, et l'ont résolu avec une nouvelle méthode de recherche multi étape nommée *Stochastic Beam Search Algorithm*. Lai et Shih (1992) proposent quelques heuristiques pour le problème, motivés par le besoin d'utiliser efficacement les zones d'accostage dans le port de Hong Kong. Ils adoptent la stratégie du *premier arrivé premier servi,* ce qui ne mène pas à un ordonnancement optimal.

Dans Imai et al. (2007) la zone d'accostage peut jusqu'à deux navires si les dimensions le permettent. Le modèle proposé est une variante hybride-dynamique du problème d'accostage avec minimisation du temps de service total résolue par les algorithmes génétiques.

Le problème d'allocation des zones d'accostage dans la plupart des travaux a été considéré comme un problème a un seul objectif. Néanmoins quelques auteurs l'ont modélisé comme un problème multi-objectif, Imai et al.(1997, 2007) , Cheong et al. (2007, 2008) et Golias et al. (2009). Imai et al. (1997) ont été les pionniers à conclure que le problème d'allocation des zones d'accostage est de nature multiobjective, ils ont considéré à l'époque la minimisation du temps de service et la minimisation de la non-satisfaction des clients par rapport au service d'accostage et de manutention (chargement/déchargement).

Les deux objectifs considérées dans (Imai et al., 2007; Cheong et al., 2008) sont alors de minimiser le temps de service et le délai dû au retard de départ des navires. La résolution dans le cas de Imai et al. (2007) a été faite par deux heuristiques, à savoir, la descente du gradient et l'algorithme génétique. Cheong et al. (2008) résolvent le problème par la méta-heuristique des colonies de fourmis. Dans leur étude, Cheong et al. (2007) considèrent 3 objectifs pour le problème, minimiser à la fois le *make-span* du port (durée entre l'arrivée du premier navire et le départ du dernier navire), le temps d'attente des navires avant d'accoster et le nombre total de *Crossing* entre les navires. Pour la résolution, ils proposent un algorithme évolutionnaire basé sur l'algorithme génétique et lui incorporent le principe d'optimalité de Pareto.

Intégration BAP-CAP

Dans cette optique, le problème visé est un problème d'ordonnancement de zones d'accostage avec allocation de ressources. La raison du choix de l'intégration de ces deux problèmes en même temps vient du fait de leur interaction réelle et effective dans un port. En effet, le but d'un *CAP* (*Crane Assignment Problem*) est de déterminer le temps de service: chargement/déchargement, qui représente en lui-même une variable d'entrée du problème de *BAP*. De ce fait, la modélisation des deux problèmes simultanément nous rapproche de la réalité portuaire, et donc, la résolution du problème combiné serait d'une applicabilité immédiate pour un gestionnaire portuaire. L'intégration des deux volets (*BAP*) et (QCAP : *Quay Crane Allocation Problem*) mène à ce qu'on appelle BACAP (*Berth Allocation and crane Assignment Problem*) et cette combinaison a commencé à retenir l'intérêt des chercheurs dans le domaine depuis déjà une décennie. Les pionniers ont été Park et Kim (2003). Ils ont modélisé le problème dans sa variante statique-continue en programmation linéaire en nombre entier (*Integer Programming*) et ont adopté une résolution en deux phases, faisant appel à une méthode heuristique basée sur la relaxation Lagrangienne pour la résolution. Meisel et Bierwirth (2006) se sont intéressés à la variante dynamique et ont classé le problème comme *Resource Constrained Project Scheduling Problem* (RCPSP). Ils l'ont résolu par une méthode heuristique basée sur des heuristiques appropriées au RCPSP.

Une variante statique discrète a été étudiée par Liang et al. (2009a et 2009b) où ils ont modélisé le problème en une approche mono-objective et multi-objective et ont adopté dans les 2 cas l'algorithme génétique pour la résolution. Imai et al. (2008) se sont penchés sur la version discrète-dynamique. Ils ont modélisé comme objectif la minimisation du temps total de service auquel ils ont incorporé les contraintes de CAP. La résolution a été basée sur la méta-heuristique des algorithmes génétiques, ils ont argumenté ce choix par l'impossibilité de résoudre ce genre de problèmes par les outils commerciaux de programmation mathématique.

Miseil et Bierwirth (2009) se sont basés sur le modèle proposé par les pionniers Park et Kim (2003) et ils ont proposé une résolution en une seule phase par une construction d'une solution faisable puis l'amélioration de cette solution par une méta-heuristique. Ils ont comparé les résultats générés par deux méta-heuristiques à savoir la recherche taboue et la *Squealy Wheel Optimization* (SWO) aux résultats donnés par Park et Kim (2003) et ont conclu que les algorithmes génétiques n'étaient pas très performants pour la résolution de tels problèmes.

Un travail original de Giallombardo et al. (2008) dans la variante discrète dynamique du problème intégré, qui utilise un modèle au départ quadratique à variable mixtes puis linéarisé présente le concept de « profiles » pour l'assignation des grues aux différents navires, ces profiles peuvent varier durant l'opération de chargement/déchargement.

Un survey de Bierwirth et Meisel (2010) s'est intéressé à la revue de littérature du problème d'intégration de BAP et CAP. Ils ont recensé les modèles posés pour le BACAP (*Berth allocation and Crane Assignment Problem*) et les méthodes de résolutions qui ont été proposées depuis cette dernière demi-décennie. Ils en ont conclu que l'intérêt est grandissant pour ce genre de problématique dans la gestion portuaire et encouragent les futurs chercheurs à trouver des modèles plus réalistes et des méthodes de résolution plus efficaces.

Cinq ans après, en 2015, les mêmes auteurs ont publié une mise à jour de cette revue de littérature (*follow up survey*) et ont constaté que l'effort continu de recherche (plus que 120 nouvelles publications depuis 2010) dans le domaine du *Berth Allocation* prouve que c'est un champ novateur et qu'il y a encore de nouvelles perspectives à explorer. Les travaux ont porté sur les méthodes de résolutions approchées par heuristiques et méta-heuristiques qui ont dominé les méthodes exactes puisque le problème a été classifié comme NP-difficile dans les 2 variantes discrète et continue. Les nouveautés ont été observées aussi du point de vue des caractéristiques des modèles présentés.

Yang et al. (2012) présentent un problème couplé entre le BAP et le CAP qui consistent en 2 sous-problèmes résolus avec un algorithme évolutionnaire basé sur des boucles imbriquées. Molins et al. (2014) dans leur problème continu dynamique pour la minimisation du temps total pondéré d'attente pour tous les navires, assignent les grues aux différentes cales du navire et prennent en considération les temps de transfert des grues entres cales d'un même navire ou entre navires. La méta-heuristique utilisée est de type GRASP et les algorithmes ont été bien détaillés dans la publication.

Raa et al. (2011) proposent un modèle enrichi linéaire à variables mixtes dans le cas continu dynamique qui autorise le déplacement des grues entre navires en cours de chargements. L'objectif est de minimiser 3 composants. Le premier est lié au temps total de chargement/déchargement, le second est lié à la positon d'accostage et inclus un terme de pénalité due à un emplacement non souhaité, et le troisième pénalise une variation positive du nombre de grues pour un navire durant son service.

Chang et al. (2010) incorporent la composante de consommation d'énergie par les grues de chargement dans la prise de décision et présentent le problème discret dynamique sous forme multi objective, transformée en monoobjective pondérée. La résolution est faite par un algorithme génétique hybride parallèle.

Hu et al. (2014) proposent un modèle non linéaire à variables mixtes, multi objectif qui tient compte de la consommation d'énergie et des émissions des grues.
Les arrivées cycliques des navires ont aussi suscité l'intérêt de quelques auteurs (Hendrinks et al. (2010), Imai et al. (2014).

Lalla-Ruiz et al. (2014) se basent sur le modèle proposé par Giallombardo (2010) et reprennent la notion de « profiles » dans l'assignation des grues aux navires, leur contribution est dans la résolution du modèle par un algorithme génétique particulier (*Biased Random Key Genetic Algorithm*) et dans la présentation d'un benchmark pour la communauté désirant travailler sur le problème.

Une autre originalité a été celle des travaux de Zhou et al. (2008) qui ont considéré l'aspect stochastique de l'arrivée des navires pour minimiser le temps moyen d'attente des navires dans le terminal. La résolution a été faite par les algorithmes génétiques.

Un aspect intéressant relevant de la réalité portuaire est celui d'autoriser les déplacements de grues entre navires durant les opérations de chargement/déchargement. Parmi les travaux qui ont considéré cet aspect, on peut citer ceux de Park et Kim (2003), Meisel et Bierwirth (2009) et Zhang et al. (2010). En effet, au lieu que les grues soient assignées à chaque navire sans fluctuations (pas de changement du nombre de grues au cours de l'opération de chargement/ déchargement), une certaine flexibilité est autorisée aux grues assignées pour se déplacer entre les navires en cours de service comme le montre la figure 0.9b.

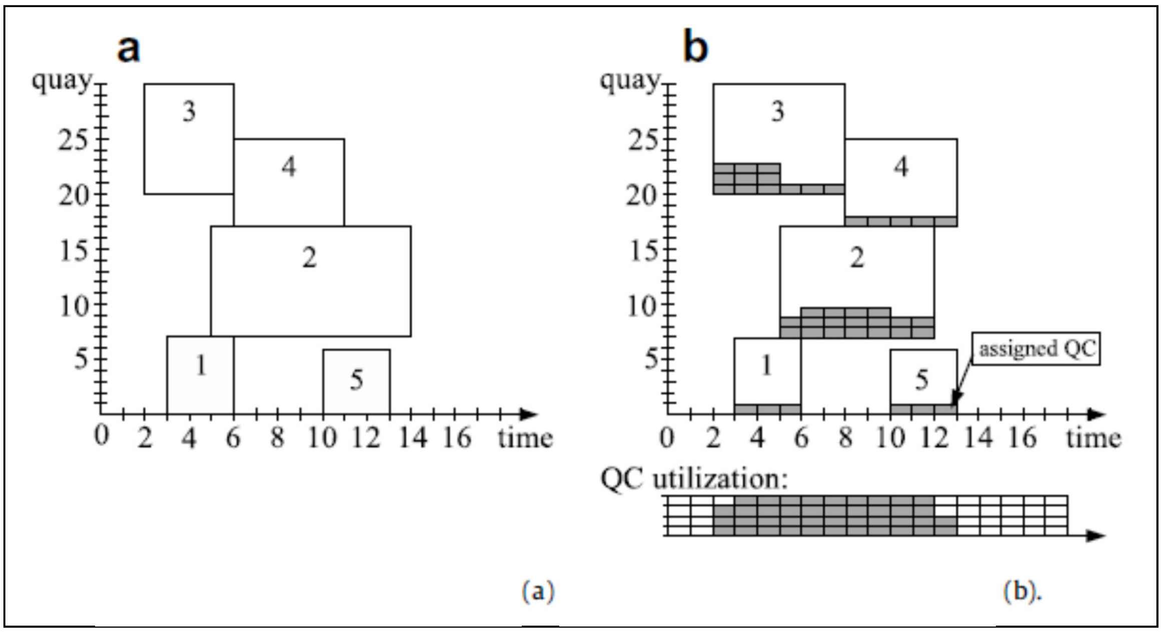

Figure 0.9 – BACAP classique (a) vs BACAP avec assignation variable de grues (b)

Cet aspect a suscité l'intérêt de plus en plus de chercheurs surtout après le denier *Survey* de Bierwirth and Miesel (2015). Dans l'état de l'art du chapitre-article 3, quelques récents travaux (datant des 2 dernières années 2016 et 2017) portant sur la stratégie d'assignation variable de grues ont été survolés.

Structure de la thèse

Les contributions de ce travail de recherche sont présentées à travers trois (3) articles de revues, qui constituent les trois (3) chapitres de cette thèse. Le premier chapitre revisite un modèle non linaire trouvé dans la littérature et pris comme référence pour la première partie du travail. Une méta-heuristique de recherche locale lui a été appliquée et une nouvelle variante multi-objective de cette approche a été proposée et appliquée pour la résolution d'un problème multi-objectif pris aussi de la littérature.

Les deuxième et troisième chapitres traitent le cas du terminal à conteneurs tunisien de RADÈS. Deux nouvelles modélisations mathématiques traduisant une partie de la réalité des opérations de planification relatives aux problèmes de *Berth Allocation and Crane Assignment* sont proposées, adoptant chacune une stratégie d'assignation de grues. Les deux modèles ont été résolus via des heuristiques et méta-heuristiques.

Une conclusion générale récapitule ce travail et des perspectives sont lancées, notamment par rapport à d'éventuelles prises en compte d'autres aspects de la réalité du terminal de Radès pour bâtir d'autres modèles mathématiques.

En annexe le cas du terminal à conteneurs de Radès est présenté pour formuler le besoin ressenti par les gestionnaires par rapport à la problématique de notre thèse, soit the *Berth Allocation and Crane Assignment Problem.*

CHAPITRE 1

EXTENDED GREAT DELUGE METAHEURISTIC BASED APPROACH FOR THE INTEGRATED DYNAMIC BERTH ALLOCATION AND MOBILE CRANE ASSIGNMENT PROBLEM

El Asli Neila[1], Dao Thien-My[1], Bouchriha Hanen[2]

[1]Mechanical and Industrial Engineering, École de Technologie Supérieure (ETS)

Montreal, Canada,

[2]Industrial Engineering, National Engineering School of Tunis (ENIT),

This chapter has been published in the « International Journal of Advanced Engineering Research and Applications », Volume – 2, Issue – 5, September – 2016

1.1 Abstract

In order for terminals to accommodate the growth in International container transport, they must make significant changes to maintain their position with increasing demand. One important manner in which existing terminal capacity could be increased would be through more efficiency. In this paper, we consider terminal efficiency from the perspective of simultaneously improving both berth and quay crane scheduling. The approach is applied to a discrete and dynamic berth allocation and crane assignment problem for both mono-objective and multi-objective variants. The problem is solved through a neighborhood meta-heuristic called the Extended Great Deluge (EGD). The results obtained with this meta-heuristic have shown better results than a Genetic Algorithm proposed in other works. A Simulated Annealing algorithm (SA) is also implemented to serve as basis of comparison for new instances results. Both algorithms (EGD and SA) for mono-objective variant have been applied to different size instances based on real world and generated data. Two new EGD-based multi-objective approaches have been proposed. Computational results are presented and discussed.

Keywords: Berth Allocation Problem (BAP); Container terminal; Crane Assignment Problem, Extended Great Deluge (EGD); Meta-heuristic; Muti-objective optimization; Simulated Annealing (SA).

1.2 Introduction

Container terminals are the areas where containers are transported from one point to another one using different handling equipments. Such terminals are continually growing in importance as maritime transport faces the challenge of using new technologies to build larger and larger ships. Moreover, transport frequency is only rising as commercial exchanges are developed to meet economic growth. To be able to compete within this environment, container terminals must be managed efficiently. To that end, managers must concentrate on the Berth, which is the most critical resource for determining container terminal capacity. An alternative approach to increasing Berth capacity involves improving its productivity through its efficient use (Park and Kim, 2003). One of the components of such efficient utilization is a focus on quay cranes, which are the main equipment used to move containers at terminals.
More and more studies are being dedicated to the examination of container terminals and efficient operations, which improve their productivity. Among them, studies dealing with berths and cranes are increasingly interest to more and more researchers.

Recently, other studies have examined the two problems simultaneously, because they are actually encountered and do interact in a container terminal. In fact, the goal of a Crane Assignment Problem (CAP) is to determine the total time of docking at the quay (including the time of service: loading/unloading and waiting time), which represents an input of the Berth allocation problem (BAP). Modeling both problems simultaneously thus approximates the reality of the harbor; consequently, resolving the joint problem would allow immediate application by a harbor manager. The combination of both the BAP and the CAP leads to an interesting problem called the BACAP (Berth Allocation and Crane Assignment Problem); this combination attracts more and more the interest of researchers in the field. The concept was pioneered by Park and Kim (2003), who modeled the problem in its static-continuous variant as an Integer Programming model, and adopted a two-phase resolution approach. The term static refers to static handling time problem, where vessel handling times are considered as input parameters whereas and by analogy the term dynamic refers to dynamic handling time where handling times are considered as decision variables since the number of cranes are also decision variables. The discrete versus continuous problems refer to the topology of the quay where in the discrete variant the quay is viewed as a finite set of berths whereas in the continuous ones, vessels can berth anywhere along the quay.

Meisel and Bierwirth (2006) were interested in the continuous-dynamic variant, and they classified the problem as a Resource Constrained Project Scheduling Problem (RCPSP). A discrete dynamic variant was studied by Liang in a mono-objective (Liang and al., 2009a) and multi-objective form (Liang and al., 2009b). They modeled the problem in a very simple and comprehensive way and adopted the genetic algorithm for the resolution. Imai et al. (2008) focus on the discrete-dynamic version. Their modeling objective was the minimization of the total time of service, including the constraints of the CAP. The resolution was based on the genetic algorithm, which is considered among the dominant algorithms proposed in the literature to solve such problems. This finding is presented by Bierwith and Meisel (2015) in their recent follow-up survey of berth allocation and quay crane scheduling problems.

Meisel and Bierwirth (2009) used the model suggested by the pioneers (Park and Kim,2003), and proposed a one-phase resolution based on the construction of a feasible solution, which was then further improved by meta-heuristics. Bierwirth and Meisel in 2010 and 2015 were interested in the review of the literature on the integration of BAP and CAP problems. They listed the models formulated for the BACAP (Berth allocation and Crane Assignment Problem) and those used in resolutions have been proposed over the last ten years. They concluded that there is a growing interest in such problems relating to in container terminal management, and thus they encourage future researchers to find new models which should be more realistic and new effective resolution methods. According to the same authors (Bierwirth and Meisel, 2015), it's not surprising that heuristics and meta-heuristics approaches dominate the resolution approaches in the literature since the berth allocation problem is known to be NP hard. Among the heuristic approaches genetic algorithm take the largest part. We can find other meta-heuristics like Tabu Search (Cordeau and al., 2005), Ant colony (Cheong and Tan,2008), Simulated annealing (Kim and Moon, 2003). Another set of studies have adopted specific heuristics like local search (Lee and chen, 2009) and greedy rules (Lee and hen, 2009). According to Bierwirth and Miesel (2015), specific local search algorithms, meta-heuristics, especially mathematically driven heuristics and exact methods have been under represented so far. They have also conclude that GA are used for their ease of implementation, however, these approaches are often rough and limited in regards to solution quality. This is the reason for our choice for local search metaheuristics. Infact, for this specific model, this work tries to propose a specific heuristic and neighborhood search method to improve the solution quality.

In this paper and as mentioned above, the discrete dynamic variant of the simultaneous berth allocation and crane assignment problem presented by Liang and al. (2009a and 2009b) will be taken and solved by means of both the Extended Great Deluge algorithm and the simulated annealing. Both methods use simple Neighborhood search heuristics to obtain near optimal solutions in a practical use and within a relatively short amount of time. A comparison between the results of the two algorithms shows the efficiency of the EGD.

In the next section, the mathematical models for both mono-objective and multi-objective variants are presented and detailed. In section 3 the methodology adopted to solve the mono-objective variant by the EGD and SA is explained, which is the first contribution in this study. In section four, and because the EGD present the best results for the mono-objective variant, two new multi-objective algorithms based on the Extended Great Deluge meta-heuristic and the dominance principle are proposed and this represents a second contribution in this paper.

1.3 BACAP Presentation

1.3.1 Liang's problem

Among several BACAP problems encounter in the literature, we consider the one presented by Liang and al.(2009a and 2009b) in its discrete-dynamic variant. The problem is chosen for its simplicity of comprehension and because it is inspired from a real container terminal in China. The first objective (2009a), presented as the total time minimization makes the model very generalizable, and capable of being applied to most container terminal situations.

The authors approached the problem to determine the exact position and the berthing time of each ship arriving at the quay of a port, as well as the exact number of Quay Cranes assigned to each of them in order to minimize the total time of berthing to the quay. This includes:

the time of loading/unloading, waiting time between arrival time and starting service and the time associated with the difference between the end of the service and the time of departure of the container ship estimated and programmed by the managers (Fig. 1.1).

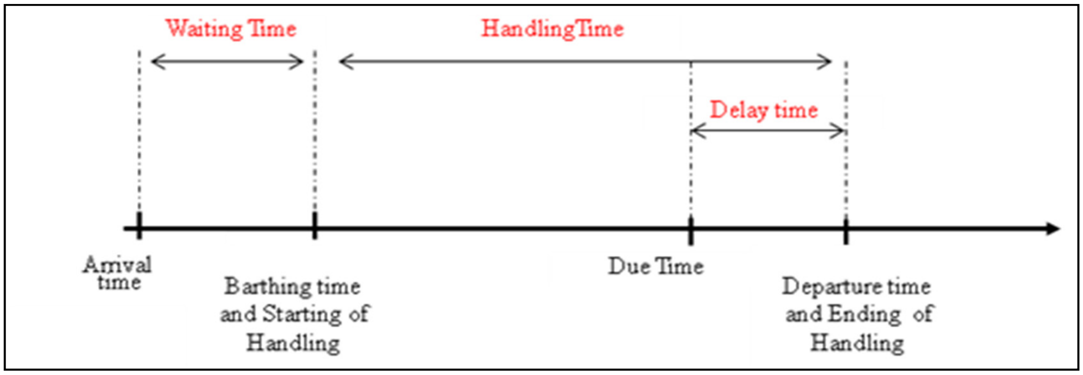

Figure 1.1 – Berth Operation timeline

Their second objective (Liang and al., 2009b) is the minimization of the workload standard deviation of cranes, considered an indicator of the efficiency of the terminal. This objective guarantees a balanced crane assignment between berths.

The assumptions below were advanced for the problem:

- Each container ship has a maximum number of cranes to be assigned.
- The time service of a container ship is directly dependent on the number of cranes assigned
- It is assumed that the time of arrival of the ship container to the port is known in advance, but the ship cannot berth before the expected arrival time.
- Loading/unloading operations must be carried out without interruption.
- Each zone of accosting must be able to accommodate a maximum of one container ship.
- The crane transfer time is ignored.

1.3.2 Problem Formulation

The mathematical model for the discrete-dynamic berth allocation and assignment problem proposed (Liang and al., 2009a and 2009b) is presented below.

We define the following indices, parameters and decision variables to formulate it:

Indices:
i (= 1, 2…n) ϵ V set of ships
j (= 1, 2…m) ϵ B set of berths
k (= 1, 2…n) ϵ O set of service orders

Parameters:
n : number of ships
m : number of berths
ν : working speed of the cranes
b : the maximum number of quay cranes that can be assigned to each ship
H : the total number of cranes available in the port
a_i : arrival time for ship i
c_i : number of containers required for loading/unloading of ship i
d_i : departure time for ship i.

Decision variables:

s_i : starting time for serving the ship i
h_j : number of cranes assigned to berth j
A : the average working time of cranes
U_j are the working times of cranes on berth j.

$$x_{ijk} \begin{cases} 1 & \text{if if the ship } i \text{ is served as the } k^{th} \text{ ship at the berth } j \\ 0 & \text{otherwise} \end{cases}$$

$$Min\ Z_1 = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{n}\frac{c_i}{\nu.h_j}x_{ijk}{}^{(*)} + \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{n}(s_i - a_i)x_{ijk}{}^{(**)} + \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{n}(s_i + \frac{c_i}{\nu.h_j} - d_i)x_{ijk}{}^{(***)} \quad (1.1)$$

$$\text{Min } Z_2 = \sqrt{\frac{1}{H}\sum_{j}^{m} h_j \left(U_j - A\right)^2} \quad (1.2)$$

Subject to

$$\sum_{j=1}^{m}\sum_{k=1}^{n} x_{ijk} = 1 \qquad \forall i \in (1,2....n) \quad (1.3)$$

$$\sum_{i=1}^{n} x_{ijk} \leq 1 \qquad \forall j \in (1,2....m), \forall \mathrm{k} \in (1,2,.....\mathrm{n}) \tag{1.4}$$

$$h_j \leq b \qquad \forall j \tag{1.5}$$

$$s_i \geq a_i \qquad \forall i \tag{1.6}$$

$$\left(s_i + \frac{c_i}{v.h_j}\right) x_{ij,k-1} \leq s_l x_{ljk} \qquad , \quad \forall \mathrm{i,k,l} = \mathrm{i}+1, \quad \forall j \in (1,2....m) \tag{1.7}$$

$$\sum_{j=1}^{m} h_j \leq \ \mathrm{H} \tag{1.8}$$

$$h_{\mathrm{j}} \qquad \text{integer } \forall \mathrm{i} \tag{1.9}$$

$$\mathrm{U_j} = \sum_{i=1}^{n}\sum_{j=1}^{m}\ \sum_{k=1}^{n} \frac{c_i}{v.h_j}.x_{ijk}, \qquad \forall j \tag{1.10}$$

$$\mathrm{A} = \frac{1}{\mathrm{H}} \sum_{\mathrm{j=1}}^{\mathrm{m}} \mathrm{h_j.U_j} \tag{1.11}$$

First, the non linear objective function (1.1) is to minimize the sum of the handling time (*) of containers for the corresponding ship, the waiting time between the arrival and the service's starting (**) and finally the delay time for every ship (***). The objective (1.2) is minimizing the workload standard deviation of cranes. Constraint (1.3) assures that every ship must be served at some berth in any order of service. Constraint (1.4) indicates that a ship must be served ones and exactly one at any berth. Constraint (1.5) restricts the maximum number of cranes used on each ship. Constraint (1.6) ensures that ships are served after their arrival. Constraint (1.7) guarantees that the handling of a ship starts after the completion of handling of its immediate predecessor at the same berth. Constraint (1.8) indicates that each crane on berth could be assigned. Constraint (1.9) enforces the number of cranes allocated to a ship to be an integer. Constraints (1.10) and (1.11) define working time and average working time between berths.

After having presented the problem formulation above, in the following section, the emphasis will be on the first objective. A new effective approach based on a local search will be presented, experienced and tested. Thereafter, in the section 1.5, the problem in its multi-objective form will be considered.

1.4 Resolution methodology for the mono-objective problem

As presented above, the Liang's model represents hard constraints that make the resolution by meta-heuristics meeting much of non-feasible solutions that the algorithm must circumvent. For this reason, a population method such as the genetic algorithm is not fully appropriate for such problems.

In this paper, and to mitigate the obstacle above mentioned, we propose to solve Liang's BACAP with a new meta-heuristic method based on neighborhood search. The Extended Great Deluge meta-heuristic is then applied. Prior to that, a heuristic is constructed to find the first feasible solution, which is gradually improved with the exploration of the neighborhood by the metaheuristic algorithm. This is what differentiates the approach suggested in this research from the resolution suggested in (Liang and al., 2009a), which sets on a random initial solution. Moreover, the construction of the initial feasible solution aims to increase the rate of acceptance of the meta-heuristic, which results in increasing the efficiency and speed of the resolution.

Besides the application of another type of algorithm to solve the problem, our approach allows to integrate the priority aspect as a decision strategy for the user. In fact, unlike Liang's approach, we add constraints relatively to the priority service in case of arbitrage between two arrivals. The approach suggested in this paper is to adopt the First Come First Serve (FCFS) rule. In case of arbitrage between two arrivals or more, the user can choose between the following rules: the “Most charged First” rule, “Less Charged First” rule and finally “Earliest Delivery Date” rule. Such a context could arise in order to satisfy some customers. The harbor manager could then have different scheduling scenarios and decide which strategy to adopt.

1.4.1 Construction of Initial Solution Heuristic

Before the application of the meta-heuristic, a heuristic is constructed to find the first feasible solution, which is gradually improved with the exploration of the neighborhood by the meta-heuristic algorithm. This is what differentiates the approach suggested in this research from the resolution suggested in (Liang and al., 2009a), which sets on a random initial solution. In general, it is expected that the better the initial solution, the better the final solution.
At the beginning the ships are affected randomly to the berths, satisfying the constraint of the arrival times, and then the cranes are assigned randomly to the berths. Once the calculation of times done, a test is launched to make sure that at any moment the total number of cranes used does not exceed the total number available. If the test fails, a modification to some crane assignments is allowed. A recalculation of the starting times, the waiting times, the delay and the total time is then performed.

The result is the initial solution that will be gradually improved by the meta-heuristic. The heuristic is repeated until obtaining an upper bound Total Service specified by the user. Figure 1.2 presents the several steps of the initial solution construction heuristic.

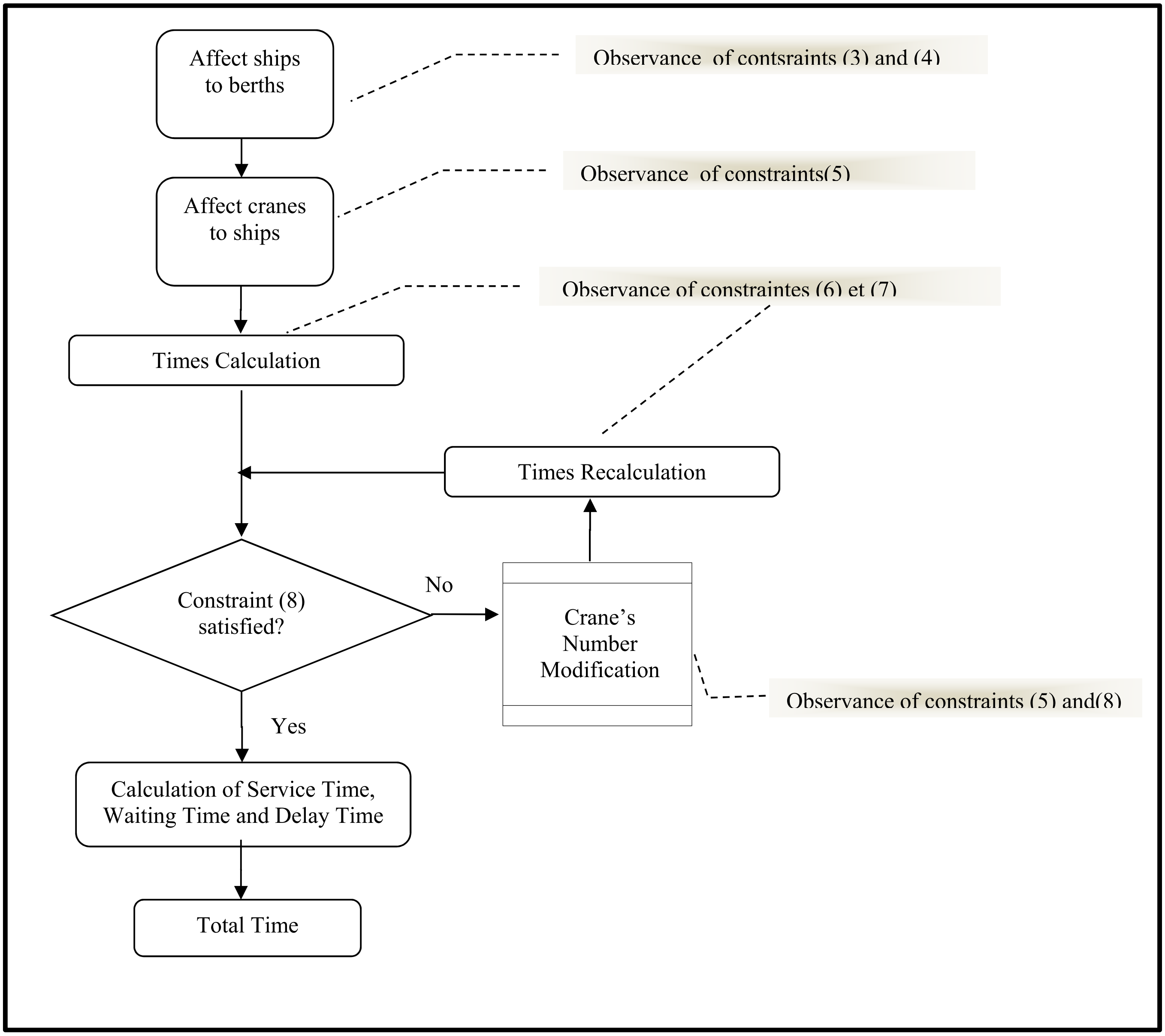

Figure 1.2 – Initial Solution Construction Heuristic

1.5 Extended Great Deluge meta-heuristic Vs Simulated Annealing

As explained above, the choice of the meta-heuristic that will improve the initial solution was related to a local search or neighborhood meta-heuristics. To that end, we explored the relatively new Extended Great Deluge (EGD) algorithm (Burke and al., 2004) and compared the results to those of the most popular local search meta-heuristic, the Simulated Annealing

The EGD algorithm based on a neighborhood search accepts every solution whose objective function is less than or equal to an upper limit (level) B or less than a current solution. The value of B is monotonically decreased during the search and bounds the feasible region of the search space. The advantage of this method is that only one input parameter, called ΔB (cf the following pseudo code in Algorithm 1.1), which is the decay rate at each step, has to be adjusted. According to Burke and al.(2004) , founder of the method, this parameter can be interpreted as a function of expected search time and expected solution quality, which are relatively easy to specify.

For the application of this algorithm to our problem, we needed:

- The initial solution S found by the constructed heuristic.
- The definition of the neighborhood N(S) of this solution.
- The neighborhood was created while making minor modifications to the initial solution S, such as to the permutation between two container ships taken randomly.
- The permutation was done for both the berth and the cranes assignment. Following the modifications, the algorithm applied tests on the neighborhood solution to check if all the constraints have been fulfilled.

Improvement regarding initial solution is carried out through implementation of the EGD algorithm presented in Algorithm 1.1. As mentioned above, it uses a boundary B, which is initially set equal to the initial solution, and is reduced gradually through the improvement process.

Besides the EGD implementation, a simulated annealing (SA) algorithm is used too in this paper to compare the results for new instances. The same heuristic for the initial solution construction is considered before its improvement by the simulated annealing algorithm. In Algorithm 1.2, the famous traditional SA is described.

Algorithme 1.1 – Extended Great Deluge algorithm

Set the initial solution ***S***
Calculate initial cost function ***f(s)***
Initial ceiling ***B=f(s)***
Specify input parameter Δ***B=?***
While not stopping condition do
Define neighbourhood ***N(s)***
Randomly select the candidate solution S^* Є ***N(s)***
If $(f(s^*) \leq f(s))$ or $(f(s^*) \leq B)$
Then Accept S^*
Lower the ceiling $B = B - \Delta B$
End while.

The simulated annealing is also a neighborhood search probabilistic meta-heuristic which emulates the physical annealing process in metallurgy. In the SA, The acceptance criterion with the probability p(T,s,s*) is employed between successive iterations.

Here, the candidate solutions with less objective function values than the current one are accepted with the probability p(T,s,s*) ,where T is a parameter called the « temperature » which is usually gradually reduced during the search. The reduction scheme that is employed is known as the « cooling schedule ». Table 1: Extended Great Deluge Algorithm.
We can conclude, then, that the SA is using more than one control parameter, which makes it less convenient to use by comparing it to the EGD. According to Burke and al. (2004), «…a greater number of poor quality solutions are generated though the use of inappropriate parameter values for the SA… Although both methods can have approximately the same values of the cost function for the best results, Simulated Annealing can reach it only with properly defined parameters, while Great Deluge does it always. ». For more details about the SA algorithm, we refer the reader to (Kim and Moon, 2003).

Algorithme 1.2 – Simulated Annealing algorithm

```
Set the initial solution S
Set the initial temperature T?
Calculate initial cost function f(s)
Specify input parameter ΔT=?
While not stopping condition do
    Define neighborhood N(s)
    Randomly select the candidate solution S* Є N(s)
    Randomly select r ϵ [0,1]
p(T,s,s*) = exp ( - ( f ( s *)-f ( s ) ) / T )
          If  r  ≤  p(T,s,s*)
          Then Accept S*
          Lower the temperature T = T - ΔT
End while.
```

1.6 Priority Rules included in the model

In this paper, in comparison with Liang et al. (2009a) approach, we wish to include more priority rules.

1. Like Liang's approach: First Come First Served (FCFS) rule.
   This constraint is modeled as follows:

$$a_i x_{ijk} < a_l x_{lj(k+1)} \forall i,k \qquad \forall j \tag{1.12}$$

2. FCFS rule like (1.12) and when 2 ships assigned to the same berth have the same time arrival, we prioritize the Most charged one.

$$\begin{aligned} &\text{if } a_i x_{ijk} = a_l x_{lj(k+1)} \forall i,k \quad \forall j \\ &\text{then } c_i x_{ijk} > c_l \; x_{lj(k+1)} \forall i,k \quad \forall j \end{aligned} \tag{1.12i}$$

3. FCFS rule like (1.12) and when 2 ships assigned to the same berth have the same time arrival, we prioritize the less charged one.

$$\begin{aligned} &\text{if } a_i x_{ijk} = a_l x_{lj(k+1)} \forall i,k \quad \forall j \\ &\text{then } c_i x_{ijk} < c_l x_{lj(k+1)} \forall i,k \quad \forall j \end{aligned} \tag{1.12ii}$$

4. FCFS rule like (1) and when 2 ships assigned to the same berth have the same time arrival, we use the Earliest Delivery Date (EDD) rule.

$$\text{if } a_i x_{ijk} = a_l x_{lj(k+1)} \forall i, k \quad \forall j \tag{1.12iii}$$
$$\text{then } d_i x_{ijk} < d_l x_{lj(k+1)} \forall i, k \quad \forall j$$

1.7 Experiments and computational results

In the following, several experiments have been performed. To begin, solution in (Liang and al., 2009a) found by the genetic algorithm is compared to our EGD result. Then, and to try to generalize the EGD performance, we have taken data from benchmark (Park and Kim, 2003), which provide different size problems ranging from small to large instances. Another work (Ursavas, 2014), presenting useful inputs to our BACAP has been taken as another source data.

1.7.1 Comparison with Liang's approach

The aim of this section is to compare the results obtained with our approach with those obtained in (Liang and al., 2009a), where the authors applied their method to solve a real case, coming from one of Shanghai container terminal companies in China. In that case, there were 4 berths and 7 quay cranes. The working speed of quay crane is common to all the cranes and was set to 40TEU/h. The data concerning the arrival time, the due time and the capacities of the ships are shown in the Table 1.3. They represent a-one day real case data from one of Shanghai container terminal companies in China, for 11 ships/day.

After finding a feasible solution by the initial solution construction heuristic, we dealt with the EGD parameters tuning to ensure good quality of the final solution. The highlight of the EGD is that it has only two parameters to adjust, which are the number of iterations $N_{iterations}$ and the step ΔB. For the latter, Burke and al., (2004) suggests, the formula (1.13) to calculate it, if some information about the range of possible result is available.

$$\Delta B = \frac{S_0 - f(s')}{N_{iterations}} \tag{1.13}$$

where f(s') is the cost function of a desired result and S_0 is the initial solution.

Tableau 1.1 – Liang's Ship informations

	Ship name	**Arrival Time**	**Due Time**	**Total number of container loading/unloading (TEU)**
1	MSG	09:00	20:00	428
2	NTD	09:00	21:00	455
3	CG	00:30	13:00	259
4	NT	21:00	23:50	172
5	LZ	00:30	23:50	684
6	XY	08:30	21:00	356
7	LZI	07:00	20:30	435
8	GC	11:30	23:50	350
9	LP	21:30	23:50	150
10	LYQ	22:00	23:50	150
11	CCG	09:00	23:50	333

In our case, Liang and al. (2009a) have tried to find a near-optimal solution to the problem and hence f(s') is fixed.

For the parameters' tuning we adopt (1.13) for different $N_{iterations}$ (1000, 2000, 5000 10 000, 20 000, 30 000, 50 000 and 100 000). For each combination, the EGD is applied several times. The best results for the 11 ships instance, presented in Table 4, are provided with the parameters $N_{iterations}$ = 50 000 and $\Delta B = 5.10^{-4}$. Our best solution is 90 minutes less than Liang's one, which represent 4% of improvement. We also found several solutions which are lower than the near-optimal one (2165 minutes) found in (Liang and al., 2009a) which could suggest that the EGD outperforms the Genetic algorithm to find a better solution for this specific problem and for this size of instances.

Tableau 1.2 – Optimal Solutions for Liang's 11-ships instance

Time in minutes	Liang's Solution (11 ships)	EGD Solution (11 ships)	improvement
Total Service Time	2165	2076.5	4%
Handling Time	1555	1563.6	
Waiting Time	610	512.9	
Delay	0	0	

The EGD solution provide as well as the GA solution no delay time, a handling time slightly higher than the AG, but a waiting time and consequently a total service time significantly lower than the genetic algorithm. This solution is more interesting for the customer finding the service more satisfying.

1.7.2 Comparaison with Benchmark data set.

Unfortunately, in Liand and al. (2009a), the authors did not provide other instances with different sizes. This is the reason why, to generalize our approach, and in addition to the real case instance with 11 ships (Table 1.1), we use data from a benchmark proposed in (Park and Kim, 2003) (for another kind of BACAP) from where we used 23 real case problems between 13 and 40 ships during the planning horizon. Of course, the benchmark was further expanded in terms of their model's data, this is why, we simply used some data that are useful to us. We have also been forced to convert service time assumed as input in their model into a number of containers (by a simple multiplication by the cranes 'speed).

To verify the performance of our EGD, and because we do not have near–optimal solutions by others meta-heuristics to compare the results, we solved these real case problems by an implemented a traditional simulated annealing (SA) algorithm which uses the same initial solution construction heuristic. We compare the results of the two meta-heuristics to highlight the advantage of using the EGD.

Another interesting instance, from (Ursavas, 2004), has been also used and solved by both EGD and SA. Table 1.3 shows the input data extracted from (Ursavas, 2014) data set. There are 21 ships, 4 berths, 7 cranes and one week horizon plan.

For the computation, we took the same crane speed for all the instances and set it to 40TEU/hour.

Tableau 1.3 – Ships instance' Informations

URSAVAS	Ship name	Arrival Time (hour)	Due Time (hour)	Total number of container loading/unloading (TEU)
1	MSC1	6.25	31	330
2	NPT	17	42	873
3	MRS1	24.5	49	358
4	HMS1	24	49	517
5	WW1	25.833	50	122
6	KPE1	26.416	51	621
7	VDB1	32.833	57	210
8	ORK1	41.5	66	1336

Tableau 1.3 (suite)

URSAVAS	Ship name	Arrival Time (hour)	Due Time (hour)	Total number of container loading/unloading (TEU)
9	WND1	42.5	67	380
10	LYQ1	45,833	70	349
11	MAR1	50	74	885
12	MSC2	51,25	76	214
13	MRS2	64,5	89	668
14	HMS2	72,66	97	236
15	WW2	90,75	115	1310
16	KPE2	101,5	126	573
17	VDB2	106,33	131	615
18	ORK2	111,58	136	401
19	WND2	129,75	155	608
20	LYQ2	130,25	156	130
21	MAR2	151,25	176	1830

As presented previously, 22 real cases and 6 generated cases were taken from (Park and Kim, 2003), distributed as follow: for the real cases, 4 problems with 13 ships each, 4 x 14 ships, 2 x 15 ships, 7 x 16 ships and 5 x 17 ships. For the generated ones, 2 x 20 ships, 2 x 30 ships and 2 x 40 ships.

We classify these instances according to their sizes in small (13-15ships), medium (16-17ships) and large (20-40 ships) classes. In the following tables (7, 8 and 9). We present the solutions found for the different classes of real case problems and generated ones with both EGD and SA.

Tableau 1.4 – EGD Vs. RS near-optimal Solutions for Small Class

Problem Size Park& Kim' Real cases	***Container average Window time average Inter-arrival average***	***EGD (min)***		***SA(min)***	
13_1	947 cont. /ship 18.15 h / ship 13.16 h between 2 ships	*4980*	*383 / ship*	*4980*	*383/ship*
13_2	981 cont. /ship 19.46 h / ship 13.5 h between 2 ships	5155	396 / ship	5155	404/ship
13_3	1083 cont. /ship 18.5 h / ship 13 h between 2 ships	5590	430/ship	5590	430/ship
13_4	1046 cont. /ship 19.7 h / ship 12.6 h between 2 ships	5530	425/ship	5530	425/ship
14_1	1005.7 cont. /ship 19.2 h / ship 12.2 h between 2 ships	5600	400 / ship	5600	400/ship
14_2	1051 cont. /ship 22.7 h / ship 12.15 h between 2 ships	6220	444 /ship	6220	444 /ship
14_3	1051.4 cont. /ship 19.2 h / ship 12.2 h between 2 ships	6010	429/ship	6010	429/ship
14_4	1108 cont. /ship 21 h / ship 12. h between 2 ships	6220	444 /ship	6220	444 /ship
15_1	1056 cont. /ship 20.3 h / ship 10 h between 2 ships	6130	408 /ship	6130	408 /ship
15_2	960 cont. /ship 19.2 h / ship 11.7 h between 2 ships	6490	432 /ship	6490	432 /ship

Tableau 1.5 – EGD Vs. RS near-optimal Solutions for Medium Class

Problem Size Park& Kim' Real cases	***Container average Window time average Inter-arrival average***	***EGD (min)***		***SA(min)***	
16_1	1045 cont. /ship 20.18 h / ship 10.9 h between 2 ships	6860	428/ship	6920	432/ship
16_2	1025 cont. /ship 17.8 h / ship 10.7 h between 2 ships	7210	450/ship	7210	450/ship
16_3	985 cont. /ship 18.18 h / ship 10.7 h between 2 ships	6940	433 /ship	7120	445
16_4	990 cont. /ship 19.7 h / ship 10.8 h between 2 ships	6540	408 /ship	6540	408 /ship
16_5	1175 cont. /ship 21.9 h / ship 10.7 h between 2 ships	7790	486 /ship	7850	490 /ship
16_6	1175 cont. /ship 22 h / ship 9.4 h between 2 ships	8580	536 /ship	8760	547 /ship
16_7	1135 cont. /ship 21.35 h / ship 10.7 h between 2 ships	7830	489 /ship	7840	491 /ship
17_1	1007 cont. /ship 18.29 h / ship 10.25 h between 2 ships	8430	495 /ship	8760	515 /ship
17_2	1040 cont. /ship 18.9 h / ship 10 h between 2 ships	7350	432 /ship	7430	437 /ship
17_3	997 cont. /ship 20.35 h / ship 9.8 h between 2 ships	6980	410 /ship	7120	418 /ship
17_4	1232.9 cont. /ship 34.9 h / ship 10.12 h between 2 ships	9870	580 /ship	13180	775 /ship
17_5	1181 cont. /ship 19.6 h / ship 10.25 h between 2 ships	9360	550 /ship	9430	554 /ship

Tableau 1.6 – EGD Vs. RS near-optimal Solutions for Large Class

Problem Size Park& Kim' Real cases	***Container average Window time average Inter-arrival average***	***EGD (min)***		***SA(min)***	
Ursavas ----21 ships	598 cont. /ship 24.6 h / ship 7.25 h between 2 ships	5050	240/ship	5172	246/ship
20_1	1200 cont. /ship 14.1 h / ship 7.9 h between 2 ships	12820	641 /ship	13480	674 /ship
20_2	1132 cont. /ship 13.9 h / ship 8.8 h between 2 ships	11180	559 /ship	11880	594 /ship
30_1	1306 cont. /ship 14.1 h / ship 5.48 h between 2 ships	25410	847/ship	26130	861 /ship
30_2	1293 cont. /ship 14.8 h / ship 5.75 h between 2 ships	24490	816 /ship	26790	893 /ship
40_1	1184 cont. /ship 14,17 h / ship 3.92 h between 2 ships	57460	1436 / ship	61440	1536 /ship
40_2	1066 cont. /ship 13.25 h / ship 4h between 2 ships	55240	1381/ship	58980	1499 /ship

For each size instances, and in order to verify the homogeneity of data, we have looked more closely at the ship loading average, the average stay and the average inter- arrival of ships. For the different size instances, we have noticed a consistency between the same size data, and this is due to the fact that they are of the same terminal. The problem size is presented in the first column as follows: number of ships instance number.

In the second column, the different averages are computed, such as average number of containers on one ship, the average time window in hour (time between arrival and departure of the ship) and finally the average of inter-arrival time between 2 consecutive ships in hour. This is done in order to try to see the influence of input data on the total service time.

The results shown in Tables (1.4, 1.5 and 1.6) are the best results found by each of the algorithm after several trials and after the parameters tuning. We conclude that the EGD outperforms SA in several cases for medium and large instances. Even if the two algorithms find the same solution (for small class), EGD is doing that in less iterations than SA.

The number of iterations was the same for each instance solved by EGD or SA and set between 2.10^5 (for Small Class) and $2.5.10^5$ (for Medium Class) . For large generated size problems, this number is set between $3.5.10^5$ and 4.10^5.

The advantage of the EGD is its simplicity in tuning; in fact when we have just one parameter to adjust, it is a significant factor for the meta-heuristic use. The second important factor is that EGD is escaping from local optima, which is noticed when we see the number of accepted solutions during the search.

The small class instances are relatively simple to solve. This is due to the fact that inter-arrival times are large. In fact, at its arrival, each ship finds his place almost automatically, like a puzzle. This can be seen on the near-optimal solution schedule of the 14_1 example data as presented in Fig. 1.3.

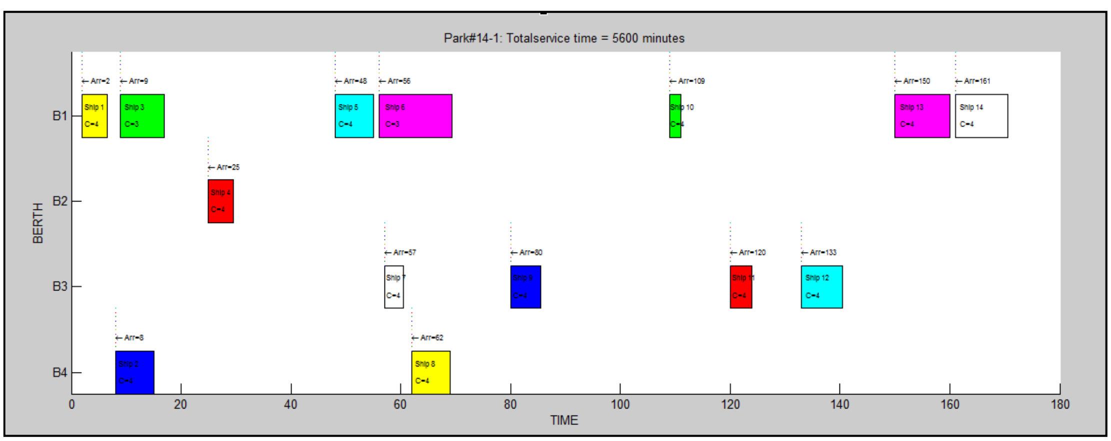

Figure 1.3 – Near optimal solution schedule for data 14_1

From Tables 1.4, 1.5 and 1.6, we can notice the important influence of the parameter inter-arrival time on the output . In fact the total service time is more sensitive to the inter-arrival time than to the service time (here the number of container on the ship). This conclusion was drawn after solving the different instances above which reveals that, for the size problem of 14, 15 and 16 ships, the maximum total service time is found for the minimum inter-arrival time average instance.

Another finding appears when we observe more closely in Figure. 1.4, the evolution of the total time depending on the size problem. In fact, we can notice that size problem has a significant effect on this total time and especially on its distribution. When the class problem is small, total time is composed almost exclusively of service time. Waiting time begins to appear in medium class problem, and for large a class, waiting time and delay are present significantly almost for all ships, and this directly impacts resolution time.

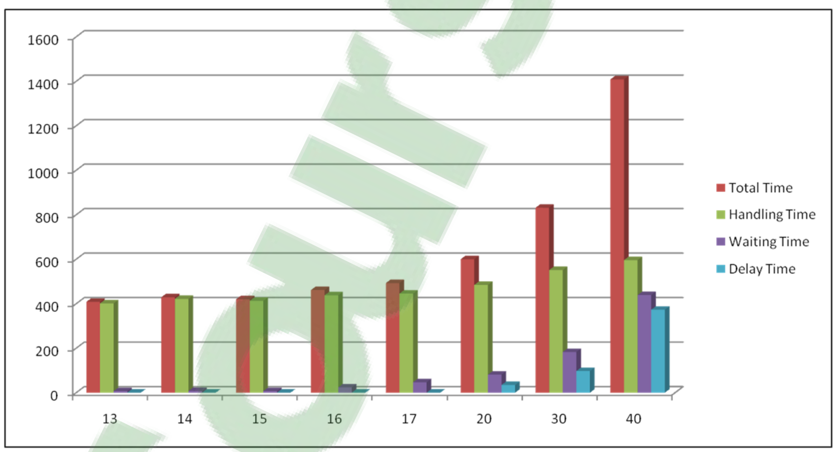

Figure 1.4 – Average Total time and its repartition Vs. problem size

Another interesting conclusion emerge in this study and this time, it is about the EGD meta-heuristic; we remark that the best results for the large instances (30 ships and more) are found when using the geometric ceil decreasing such B decreases by B*(1-ΔB) rather than B-ΔB at each iteration. This can be seen in more details in future works.

1.8 Adopted Approach to solve the multi-objective problem

At a container terminal, there are always two parts. Shipping lines who want their vessels to be served upon arrival and complete their loading/unloading operations within a prearranged time window and the terminal operators, who want to improve their efficiency, optimize their logistic process and the throughput of the terminal. Due to this fact, problems dealing with the container terminals generally have a multi-objective aspect to find a compromise between these two parts.

Berth allocation problem can be seen as a multi-objective problem where shipping lines seek to minimize their total service time and on another hand terminal operators have to offer their service in an optimal and efficient way.

As presented in part 1.2 in this work, the second objective formulated above, is the minimization of the workload standard deviation of cranes, considered an indicator of the efficiency of the terminal. For more details we refer the reader to Liang and al. (2009b).

After seeing the performance of the EGD meta-heuristic in the resolution of mono-objective BACAP, we thought to test its robustness for the multi-objective variant. Therefore, we tried in this paper to adapt the EGD to the multi-objective optimization. In the literature, to our knowledge, the only work dealing with the multi-objective great deluge algorithm was presented by Petrovic and Bykov (2002) who suggests a Multi-objective Great Deluge algorithm with Variable Weights to solve such problems. It operates with a composite objective function formed by summing the different weighted single objectives. The particularity of that sum is that the weights are varied dynamically during the search. For more details, we refer the reader to (Petrovic and Bykov, 2002).

After having implemented the algorithm presented in (Petrovic and Bykov, 2002), we still have a few remarks about some shortcomings, ie, the approach is using a transformation of the different objectives into a single one but not the classical weighted technique where having necessarily the weight sum equal to 1, It is difficult to choose the initial weight parameters which is, according to the authors important; and finally we think that it is not always possible to set a reference solution, which is, necessary to run the algorithm .

Because of the gaps in the previous algorithm, and because the EGD has not been explored thoroughly enough for the multi-objective problems, in this paper, and this is the most interesting contribution, we propose two Pareto Archived variants of EGD which were inspired from the Engrand's *Multi-Objective Simulated Anneling* (MOSA) in (Engrand, 1997) in terms of dominance principle and return to base technique. But unlike Engrand (1997) who, proposed a new function G, sum of the logarithms of the different single objective, the first variant is treating each objective separately and the second is applying the classical weighted sum, found for the most aggregated methods.

We simply called the first (PA-EGD) where each objective is evaluated separately in each iteration, then an "archive" is created to store the non-dominated solutions during the search. The second is called (PA-WEGD) because it is using weights to transform the different objectives into a single one.

These two variants are based on the non-dominance principle and thus are trying to find the Pareto set solutions. To simply summarize the non-dominance principle, let's assume that a reasonable solution to a multi-objective problem is to investigate a set of solutions, each of which satisfies the objectives at an acceptable level, and without being dominated by any other solution; it's the Pareto optimal set. Consequently, a solution belongs to the Pareto set if there is no other solution that can improve at least one of the objectives without degrading any other objective.

1.8.1 Pareto Archived EGD (PA-EGD)

This algorithm is based first on the extended great deluge acceptance for neighbour solutions and secondly on the non-dominance archiving principle. The algorithm also periodically executes a "return-to-base" option which continues search by selecting randomly a solution from the archive. This is done to try to ensure that the entire trade-off is found.

The PA-EGD starts by taking each objective i separately and associates a ceil B_i to each of them for the initialisation of the algorithm. Algorithm 1.3 presents the proposed algorithm.

In our case, B_1 is the initial total service time associated to the first solution found by the construction heuristic and B_2 is the workload standard deviation of cranes associated to this same first solution. At this stage, the meta-heuristic is launched and in each iteration, attempts to find a new solution that decreases both total service time (compared to the last total service time or the last ceil B1) and workload (compared to the last workload or the last ceil B2). The archive constituted by the non-dominated solutions among the accepted ones is then updated. An interesting way to guarantee that all the entire solution space is found is the return to base, which enables the search to restart from an archived solution.

Algorithme 1.3 – PA-EGD

Set the initial solution ***S***
Calculate initial cost functions $f_1(s), f_2(s)$
Initial ceilings $B_1=f(s); B_2=f_2(s)$
Specify input parameter $\Delta B_1; \Delta B_2=?$
While not stopping condition do
Define neighbourhood ***N(s)***
Randomly select the candidate solution $S^* \in N(s)$
If $(f_1(s^*) \leq f_1(s))$ or $(f_1(s^*) \leq B_1)$ & $(f_2(s^*) \leq f_2(s))$ or $(f_2(s^*) \leq B_2)$
Then Accept S^*
Update the Archive
do not archive S^* *if dominated by one archived individual,*
archive S^* *if not dominated by any archived individual,*
remove archived individual if dominated by S^*
Lower the ceilings $B_1 = B_1 - \Delta B_1$ *and* $B_2 = B_2 - \Delta B_2$
Periodically *« return to base »* from an archived solution.
End while.

1.8.2 Pareto Archived Weighted EGD (PA-WEGD)

In the last section, the multi-objective optimization algorithm is treating each objective separately. In this part, the search is based on an aggregating technique, which converts the different objectives into a single one by a weighted sum (Coello, 1999). The weakness of this approach s is the difficult choice of the weights in advance when we do not have enough information about the problem (Coello, 1999). To palliate to this point, we have developed an algorithm trying to cover several configurations of weights. For our case, we have 2 objectives, the weighted sum can be written then as

$$F(s) = w\, f_1(s)+(1-w)\, f_2(s) \tag{1.14}$$

We notice here that only one parameter w is adjusted. In fact, the weight w is initialized to 0.1 and is increased by 0.1 at each search iteration in order to realize various search directions to uncover more non-dominated solutions in the solution space.

Algorithme 1.4 – PA- WEGD

```
Set the initial solution S
while  0.1≤ w ≤ 0.9
  Calculate  the initial weighted function  F(s) = w f1(s)+(1-w) f2(s)
  Initial ceilings B=F(s);
  Specify input parameter  ΔB=?
    while not stopping condition do
      Define neighbourhood N(s)
      Randomly select the candidate solution S* Є N(s)
        If (F(s*) ≤  F(s))  or  (F(s*) ≤ B)
              Then Accept S*
         Update the Archive
            do not archive S* if dominated by one archived individual,
.           archive S* if not  dominated by any archived individual,
.           remove archived individual if dominated by S*
        Lower the ceiling B = B - ΔB
        w=w+0.1
     end while.
end while.
```

1.8.3 Experiments and results for the Multi-objective problem

The 2 techniques PA-EGD and PA-WEGD are tested and compared for the resolution of the Multi-objective BACAP presented in (Liang and al., 2009b).
The aim of this section is to apply the two above proposed EGD based algorithms on a 13 ships instance presented in Table 12 from (Liang and al., 2009b), and compare the results obtained with the multi-objective hybrid genetic algorithm developed by the authors.

Tableau 1.3 – 13 ships Instances Information from (Liang and al., 2009b)

	Ship name	**Arrival Time**	**Due Time**	**Total number of container loading/unloading (TEU)**
1	ZHE	01:00	17:00	525
2	ZHW	01:00	17:00	515
3	ZYE	01:00	15:00	722
4	ZYW	01:00	15:00	741
5	ZX	00:00	14:00	400
6	JWH	05:30	17:30	664
7	JYD	01:30	12:00	227
8	XNT	05:30	22:00	795
9	DY	07:54	10:25	34
10	MZ	13:54	14:59	31
11	ZH	00:00	14:30	149
12	XY	15:00	22:00	236
13	YL	20:06	23:50	105

In the following Table 1.4 and on the figures 1.5, and 1.6, the Pareto solutions and the accepted solutions, found by PA-EGD and PA-WEGD are presented for both Liang's instances with 11 and 13 ships respectively. In the second and third columns of the table 1.4, the Liang results are detailed as found by (Liang and al., 2009b).

The authors choose the solutions (Z1 --- Z2) such as (55.13 --- 2.5) & (95.2 --- 2.44) as the Pareto solution closest to the ideal points in both configurations with 11 and 13 ships. As presented graphically in (Liang and al., 2009b), the ideal points for the 11 ships instance and 13 ships instance respectively are (51---1.5) and (89---1.4).

We may notice that the Pareto solutions (bold writing) found by both the PA-EGD and PA-WEGD try to reach closely the ideal points, which prove the performance of the proposed algorithms when we compare the solutions to those found by a well known multi-objective algorithm such as GA. The figures 1.5 and 1.6 (a) show the Pareto solutions (green points) and accepted solutions (blue points) during the PA-EGD search, respectively for 13 and 11 ships The space is enveloped by the two ceils (B_1=Total service time on left and B_2 = Workload on the top). The curve Pareto represents the non-dominated solutions. By analogy, Figures 1.5 and 1.6 (b) show both Pareto solutions (red points) and accepted solutions (blue points) during the PA-WEGD search, respectively for 13 and 11 ships.

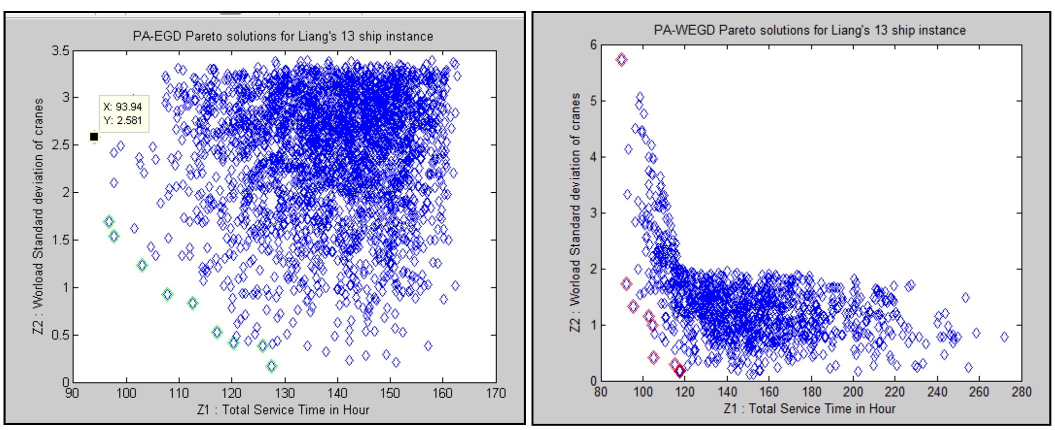

Figure 1.5 – Pareto and acceptedSolutions by PA-EGD
and PA-WEGD for 13 ships

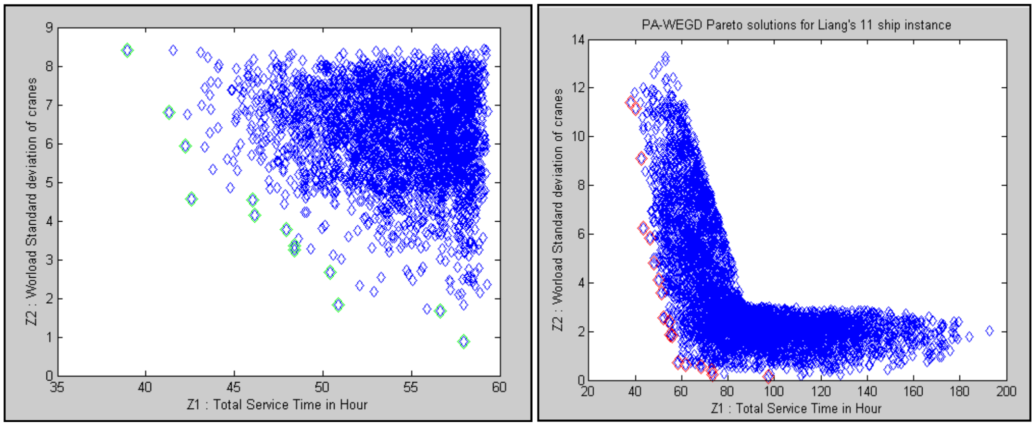

Figure 1.6 – Pareto and accepted Solutions by PA-EGD
and PA-WEGD for 11 ships

Tableau 1.4 – Multi-objective Solutions for PA-EGD & PA-WEGD

Instance	***Pareto Solution Liang (Genetic Algorithm)***		***PA-EGD***		***PA-WEGD***	
	Z_1	**Z_2**	**Z_1**	**Z_2**	**Z_1**	**Z_2**
					38.17	11.39
					40.07	11.17
			38,93	8,40	42.70	9.09
			41,33	6,79	43.59	6.24
			42,21	5,92	46.19	5.82
	51.33	4.6	42,57	4,55	47.86	4.81
	51.9	4.3	46,05	4,53	49.97	4.11
	53.15	3.12	46,15	4,14	51.38	3.57
Liang-- 11 ships	**55.13**	**2.5**	47,93	3,76	**52.15**	**2.57**
	57.68	1.8	48,38	3,34	55.84	2.41
	60.45	1.52	48,38	3,24	54.62	1.87
	62.73	1.5	50,4	2,67	55.18	1.80
			50,85	**1,83**	58.20	0.72
			56,60	1,66	61.73	0.61
			57,93	0,89	68.12	0.56
					72.84	0.38
					73.15	0.23
					97.35	0.16

Tableau 1.4 Suite

Instance	***Pareto Solution Liang (Genetic Algorithm)***	***PA-EGD***	***PA-WEGD***	***Pareto Solution Liang (Genetic Algorithm)***	***PA-EGD***	***PA-WEGD***
	Z_1	**Z_2**	**Z_1**	**Z_2**	**Z_1**	**Z_2**
Liang--13 ships	90.32	5.6				
	91.67	5.13	94.06	1.64		
	91.8	3.68	**94.38**	**1.49**		
	1.87	4.62	95.30	1.29	90,02	5,73
	93.33	3.98	103.22	0.90	**92,04**	**1,73**
	94.17	3	110.68	0.74	95,66	1,31
	95.2	**2.44**	111.01	0.71	102,74	1,14
	103.47	2.34	111.23	0.68	104,85	0,99
	114.75	1.98	111.98	0.39	105,18	0,40
	130.23	1.59	127.90	0.15	115,15	0,29
	134.70	1.48	139.33	0.12	117,67	0,19
	142.57	1.47	139.68	0.06	117,68	0,16
	152.82	1.46	113.03	0.26		
	171.48	1.36				

1.9 Conclusion

In this paper we have attempted to solve a BACAP in its discrete-dynamic variant in both mono-objective and multi-objective cases. The approach chosen is based on an extended Great Deluge meta-heuristic preceded by a heuristic to construct the initial feasible solution. The approach has exploited the inherent advantages with this Extended Great Deluge technique in escaping from local optima while also maintaining a relatively simple set of neighborhood moves. The results found in this paper could be compared to others meta-heuristics solutions.

For the mono-objective problem, the EGD results have been compared to those found by a genetic algorithm for a small real case problem and to those of Simulated Annealing algorithm implemented to solve the problem for medium and large case problems taken from an adapted (slightly revisited to fit the problem) BACAP benchmark (Park and Kim, 2003). In both cases, the performance of the algorithm has been demonstrated.

In this work, and for large instances, another way to decrease the ceil B in the EGD has been tested and proved to be efficient.

We have also concluded in this work, that for the issue treated, inter-arrival time is the most influent parameter and that the problem size affects the complexity of the resolution and thus resolution time. The two proposed algorithms developed for the multi-objective problems, named PA-EGD and PA-WEGD are subject to be more thoroughly studied for other kind of problems in the future.

CHAPITRE 2

BERTH ALLOCATION AND MOBILE CRANES TIME-INVARIANT ASSIGNMENT PROBLEM IN A SPECIAL CONTAINER TERMINAL

El Asli Neila[1], Dao Thien-My[1], Bouchriha Hanen[2]

[1]Mechanical and Industrial Engineering, École de Technologie Supérieure (ETS)

Montreal, Canada,

[2]Industrial Engineering, National Engineering School of Tunis (ENIT),

This chapter has been accepted for publication in the « Journal of Engineering and Applied Sciences », Medwell journal.

2.1 Abstract

Berth allocation and crane assignment problem (BACAP) is an interesting integration of two major problems in the container terminals ' logistic. Before (Park and Kim, 2003), many works have considered each problem separately but since 2003, researchers are increasingly interested in the combination that makes mathematical models more realistic. In this paper, we present an application of these two simultaneous problems in a particular and real case where the terminal is treating two different types of vessels simultaneously; container ships and Roll-on Roll-off (Ro-Ro) ships. A new non-linear Time Invariant Assignment model, minimizing the total service time for all vessels is presented. To solve the model, a meta-heuristic algorithm is used: the Artificial Bee Colony (ABC), a population-based algorithm never used for such problems before.

Keywords: Terminal Operations, Berth Allocation, Time Invariant Crane Assignment, Container Vessel, Ro-Ro Vessels, Artificial Bee Colony meta-heuristic

2.2 Introduction

Seaside operations planning in container terminals have a strong impact on their competitiveness. Indeed, it is considered to be the bottleneck operation in most container terminals around the world. Among modern issues of seaside operations planning is the integration of quay space and crane assignment to vessels that have to be loaded/unloaded with the aim of optimizing service time.

A large number of optimization approaches to the seaside problems has been published in the scientific literature. Recently, several studies have examined simultaneously the two problems called Berth Allocation Problem (BAP) and Crane Assignment Problem (CAP) because they do closely interact in a container terminal.

In fact, the goal of a Crane Assignment Problem (CAP) is to determine the total time of docking at the Quay (including the time of service: loading/unloading and waiting time), which represents an input of the Berth allocation problem (BAP). Modeling both problems simultaneously thus approximates the reality of the harbor; consequently, resolving the joint problem would allow immediate application by a harbor manager. The combination of both the BAP and the CAP leads to an interesting problem called the BACAP (Berth Allocation and Crane Assignment Problem); this combination increasingly attracts the interest of researchers in the field. The concept was pioneered by Park and Kim (2003) ,and has been very much discussed. Bierwierth and Meisel (2010) connectedly reviewed, then, for the first time, the literature dealing with seaside operation planning in container terminals and follow-up in Bierwierth and Meisel (2015) an update to reflect the amount of new research in the field within the 5 years separaing the two surveys. This proves the strong increase of activity observed in this research field. Contributions range from formulating new mathematical models to innovating solving approaches. For more details about the technical specifications of BACAP, we kindly refer readers then to the two surveys where they could find more exhaustive information.

Among attribute assignment which can be extracted throughout these 2 surveys; the authors distinguish between *time-invariant* and *variable in-time* crane assignment. The former means that a constant number of cranes is used for serving a vessel throughout the entire process, while the latter means that the number of cranes change dynamically during the service process (Bierwierth and Meisel, 2015). The classification of papers according to that attribute has not been treated to our knowledge. In the present study, in table 2.1, we try to inventory some integration studies dealing with these two aspects.

Tableau 2.1 – Integration studies classified by the crane assignment strategy

Assignment Attribute	Reference
Time Invariable Assignment (TIA)	Liang and al.(2009,2010,2011), El Asli and al. (2016), Salido and al.(2011), Chang and al.(2010), Lee and al.(2010), Yang and al.(2102),Imai and al.(2008), Yavuz and al.(2014), Han and al.(2010)
Variable in Time Assignment (VTA)	Ursavas (2014), Zhang and al.(2010); Meisel and Bierwierth (2009); Rodriguez-Molins and Salido (2013), Giallombardo and al.(2010), Raa and al.(2011), Hu (2015), Hsu (2016); Vacca and al.(2013), Xiao and Hu(2015); Iris and al. (2015); Lalla-Ruiz and al.(2015)

The first assignment (TIA) will be modeled in this work inspired from a real case terminal.

In this paper, a special container terminal is considered; two different types of vessels can accost and can be served; Container's vessels, and RoRo vessels.

The quay is partitioned into berths, which leads to a discrete layout. Each type of vessels has its intended space to berth.

Ro-Ro ships are vessels that are used to carry wheeled cargoes, such as cars, trucks, semi-trailers trucks, trailers and railroad cars that are driven on and off the ship having their own wheels or using a platform vehicle.

Ro-Ro ships that accost at their intended berths are also often partially loaded also with containers. This leads to a probable use of handling resources such as mobile cranes used by container ships. This sharing of a resource leads to an operation productivity decrease.

Another distinguishing feature for this container terminal is that only Ro-Ro are constrained by a time window (arrival and departure time) while container ships arriving at the port are not faced with an imposed due date. Indeed, they are feeders (a feeder is a small vessel which makes the pre and post container transport to ports with no stop), which arrives at the container terminal carrying, exclusively, the dedicated containers.

In this paper , a new nonlinear discrete dynamic variant of the simultaneous berth allocation and crane assignment problem is presented and solved by means of Ant Bee Colony (ABC) algorithm.

In the next section, the new non linear mathematical model for BACAP is presented and detailed, which is the first contribution in this study. In section 3 the methodology adopted to solve this problem, by means of the chosen population meta-heuristic, is explained. In section 4, the results are presented and analyzed .

2.3 Mathematical Model

In this non linear model, the BACAP considers discrete typology of berths (this terminal has 7 individual berths) and dynamic temporal attribute of arrival process of vessels, which considers that vessels arrive at individual but deterministic arrival times imposing a constraint for the berth allocation , and thus according to the classification scheme proposed by Bierwirth and Meisel, (2015), this study can be classified as *disc|dyn|QCAP|(wait+tard+hand).*

To formulate the BACAP , we propose to minimize waiting, handling , and delay times for both container and Ro-Ro ships, and the delay for the Ro-Ro as presented by the following compact goal form (2.1) and explained by the berth operation timeline in figure 2.1.

$$\begin{aligned} MinZ = & \sum (waiting\ time)_{RoRo+Container\ shps} \\ & + \sum (handling\ time)_{RoRo+Container\ shps} \\ & + \sum (delay)_{RoRo} \end{aligned} \tag{2.1}$$

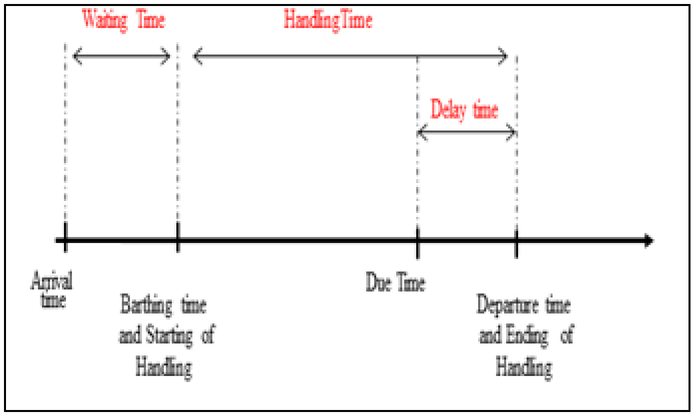

Figure 2.1 – Berth Operation Timeline

This section provides the new mathematical formulation for the Time Invariant Assignment BACAP in this special terminal.

In the following, the notations and parameters used in the model are detailed.

Input :

$i = 1, \ldots, I\ set\ of\ vessels$;

$I = \{I_{PC}\ \&\ I_{RoRo}\}$

I_{PC} and I_{RoRo} are Containers Ships & RoRo ships

$j = 1, \ldots, J\ set\ of\ discrete\ berths$; .

$j = \{1,6,7\}$ for Containers Ships

$j = \{2,3,4,5\}$ for RoRo ships

$k = 1, \ldots, K;$ set of ship services at the same berth

a_i = ship arrival time ; $i \in I$

d_i = ship departure time ; $i \in I_{RoRo}$

C_i = Ship Conaitner Capacity ; $i \in I$

R_i = Ship trailers Capacity ; $i \in I_{RoRo}$

v = Crane Speed (Cont/hour)

v_R = Truck Speed (trailer/hour)

$H = number\ of\ available\ mobile\ Crane$)

$coef_1$ = *Interference coefficient for Container handling*

$coef_2$ = *Interference coefficient for simultaneous container and trailers handling*

Decision Variables:

$$x_{ijk} = \begin{cases} 1\ if\ the\ ship\ i\ \ is\ served\ \ at\ berth\ j\ as\ the\ k^{th}\ ship \\ 0\ else \end{cases}$$

$$y_{il} = \begin{cases} 1\ if\ \ ship\ i\ is\ served\ before\ the\ completion\ time\ of\ ship\ l \\ 0\ else \end{cases}$$

$h_i = \ integer$:

$\{0,1,2\}, number\ of\ cranes\ affected\ to\ \ ship\ i, i\ \in I$

$s_i = starting\ service\ time\ of\ ship\ i, i\ \in I$

$s'_i = starting\ trailers\ hanling\ time\ of\ ship\ i, i\ \in\ I_{RoRo}$

$f_i = Finishing\ service\ time\ of\ ship\ i, i\ \in I$

$e_i = Ending\ container\ handling\ time\ of\ ship\ i, i\ \in I$

$er_i = Ending\ trailers\ handling\ time\ of\ ship\ i, i\ \in I_{RoRo}$

The mathematical non linear model is formulated as:

$$Min\ Z = \sum_i \sum_j \sum_k (s_i - a_i)\, x_{ijk} \quad + \sum_i \sum_j \sum_k (f_i - s_i)\, x_{ijk} + \sum_i \sum_j \sum_k (f_i - d_i)\, x_{ijk} \tag{2.2}$$

$$\sum_j \sum_k x_{ijk} = 1 \quad \forall\ i \epsilon I \tag{2.3}$$

(2.4)

$$\sum_i x_{ijk} \leq 1 \qquad \forall\, j\, \forall\, k$$

$$s_i \geq a_i \qquad \forall i \epsilon I \tag{2.5}$$

$$h_i \leq b_1 \qquad \forall i \epsilon I_{PC} \tag{2.6}$$

$$h_i \leq b_2 \qquad \forall i \epsilon I_{RoRo} \tag{2.7}$$

$$\sum_i s_i\ x_{ijk} \geq \sum_l \left(s_l + \frac{C_l}{v\ h_l}\right) x_{ij(k-1)} \qquad \forall j \in \{1,6,7\};\ \forall k \tag{2.8}$$

$$\sum_i s_i\ x_{ijk} \geq \sum_l max(f_l;\ d_l)\, x_{lj(k-1)} \qquad \forall j \in \{2,3,4,5\}; \forall k \tag{2.9}$$

$$h_i + \sum_{l \neq i} h_l\, y_{il} \leq H \quad \forall i \in I \tag{2.10}$$

$$er_i = s_i + \frac{R_i}{v_R}.coef_2 \ \ \forall i \in I_{RoRo} \tag{2.11}$$

$$e_i = \begin{cases} s_i + \dfrac{C_i}{vh_i}\ .coef_1\ ;\ i \in I_{PC} \\ s'_i + \dfrac{C_i}{v}\ .coef_2\ ;\ i \in I_{RoRo} \end{cases} \tag{2.12}$$

$$f_i = e_i \ ; i \in I_{PC} \tag{2.13}$$

$$f_i = max(er_i\ ;\ e_i\ ) \quad ; i \in I_{RoRo} \tag{2.14}$$

$$max(s_i\ ;\ s'_i\ ) \leq d_i \ ; i \in I_{RoRo} \tag{2.15}$$

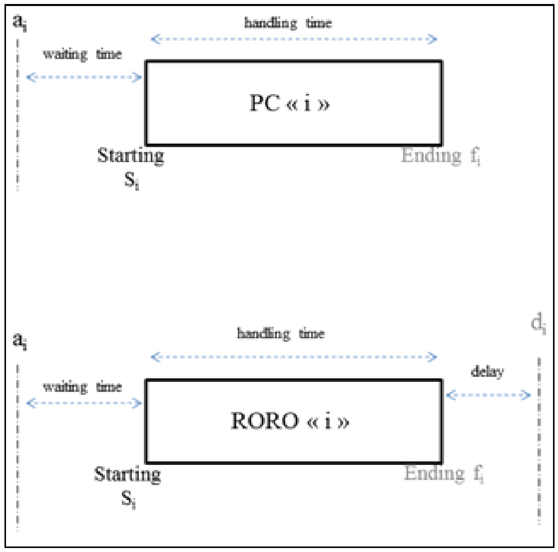

Figure 2.2 – Illustration for main RoRo parameters

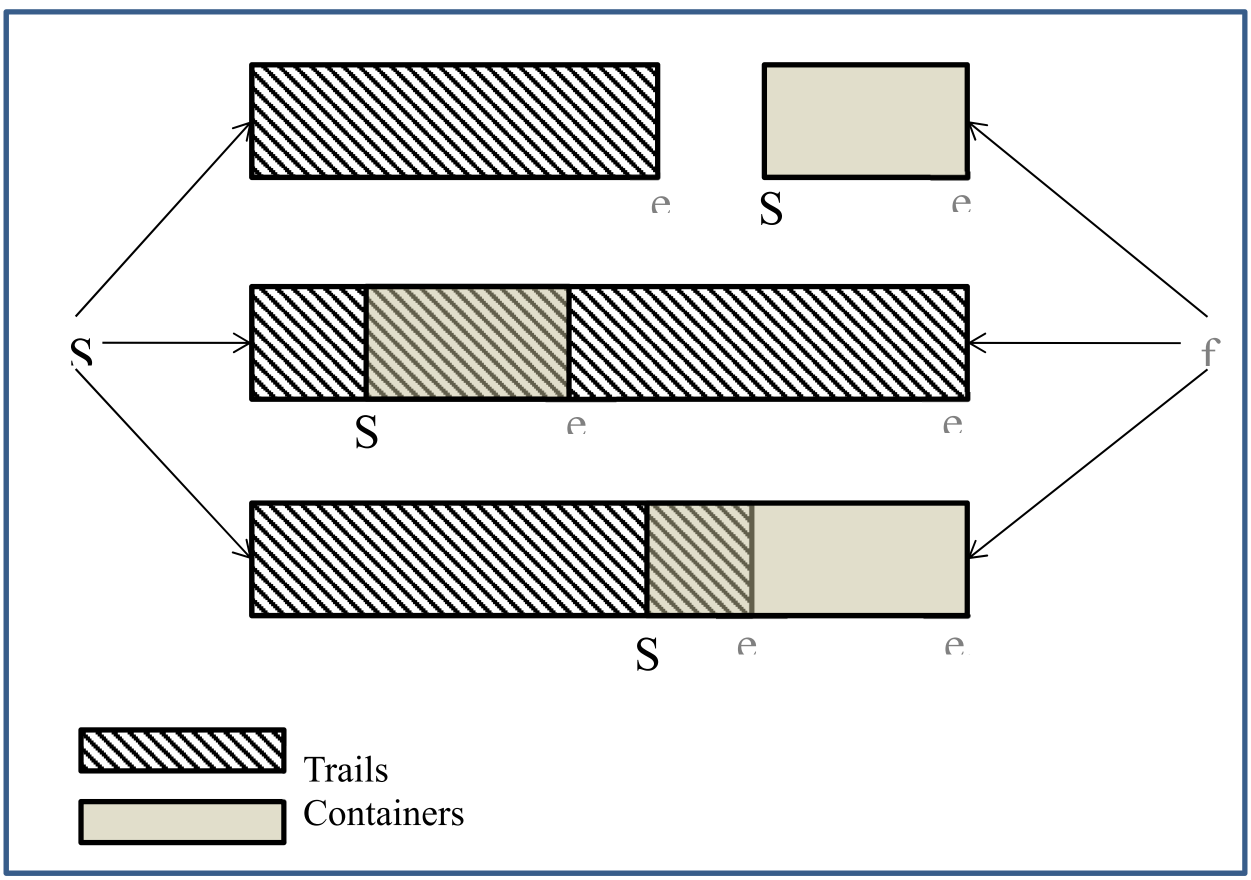

Figure. 2.3 – Different configurations of RoRo services

Before detailing the model, some assumptions should be made:

- There are intended berths for the RoRo and others for container ships.
- Only the Ro-Ro are constrained by a time window.
- The number of cranes assigned to each container ship is limited.
- Just one crane is allowed to be assigned to a Ro-Ro when necessary.
- There is a Total number of cranes in the terminal that should not be exceeded at any time.
- The transfer time is ignored when a crane new assignment is performed .
- The crane handling rate is assumed to be the same for both vessel types.

The objective function (2.2) minimizes as stated earlier, the summation of: 1- the waiting duration between the service starting moment and the arrival time, 2- the handling service duration for both types of vessels , and 3-the delay, in case if should happen (if the vessel leaves the terminal after its due time), for the RoRo vessels.

Constraint (2.3) forces that only one vessel can be served at each single berth at a time. Constraint (2.4) restricts every berth serves up to only one vessel at any time or unoccupied. Constraint (2.5) indicates that the start of the service only begins with or after ship arrival. Constraints (2.6-2.7) define the maximum number of simultaneous cranes that could be assigned to containers vessel (2 at a time max) and RoRo-vessel (only 1). Constraints (2.8-2.9) indicate that , each vessel cannot be served at any berth once its predecessor leaves. Constraint (2.10) ensures that assigned cranes do not exceed the available ones at any moment. Constraints (2.11-2.12) define the ending times for both trailers loading/unloading (er_i) and container loading/unloading (e_i). Constraint (2.13-2.14) define respectively the finishing times for the containers-ships and RoRo. Finally, constraint (2.15) enforces RoRo starting time do not exceed its departure due time.

2.4 Time Invariant Assignmnt heuristic construction

To have an initial feasible solution, the construction heuristic generates berth and crane assignments to the vessels involved throughout the planning horizon. Once the berth assignment is performed satisfying the constraints (2.3) to (2.5), the idea is to assign up to 2 cranes to container ships and just one crane to RoRo, when requested , to perform container handling operation. The hard constraint (2.10) is verified after each trial assignment, and when necessary, adjustments are performed. The heuristic tried to remove 1 crane for container ships or to shift the starting time for the crane assigned to RoRo, this can be done up to the departure time for the RoRo. The proposed framework for the construction heuristic is presented in figure 2.4.

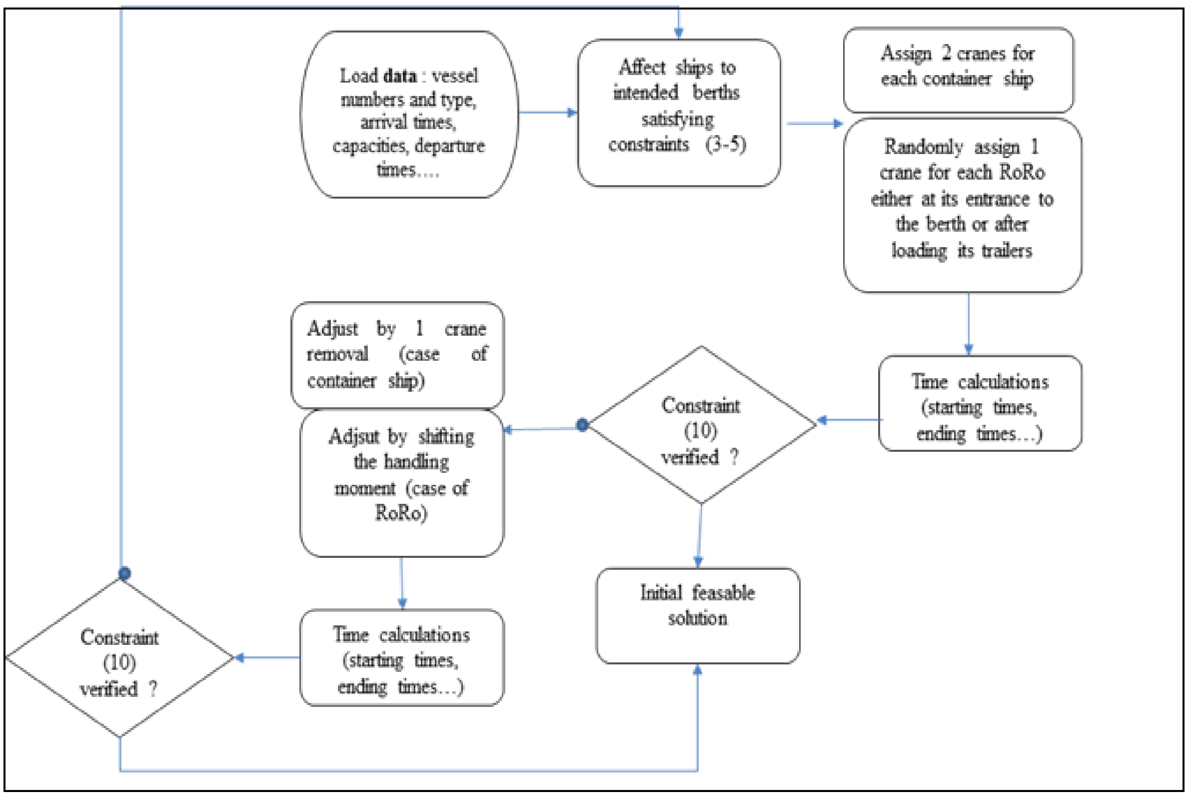

Figure 2.4 – Proposed framework for construction heuristic

2.5 Artificial Bee Colony (ABC) based approach to solve BACAP-TIA

An ABC, which is an optimization algorithm based on the intelligent behaviour of honey bee swarm (Karaboga and Basturk, 2007) is then performed. ABC is one of the most applied Swarm Intelligent algorithm to solve many different types of applications in the past few years. The ABC is a population-based algorithm for combinatorial optimization that is inspired by the foraging behavior of bees. It mimics the colony behavior to search for the best source of food.

As in the minimal model of forage selection of real honey bees, the colony of artificial bees in ABC contains three groups of bees: employed bees associated with specific food sources, onlooker bees watching and decoding the dance of employed bees within the hive to choose a food source, and scout bees searching for food other sources randomly. Both onlookers and scouts are also called unemployed bees. Initially, all food source positions are discovered by scout bees.

Thereafter, the nectar of food sources is exploited by employed bees and onlooker bees, and this continual exploitation will ultimately cause them to become exhausted. Then, the employed bee which was exploiting the exhausted food source becomes a scout bee in search of further food sources once again.

In ABC, the position of a food source represents a possible solution (initial feasible solution) to the problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. In the basic form, the number of employed bees is equal to the number of food sources (solutions) since each employed bee is associated with one and only one food source.

The general algorithmic structure of the ABC optimization approach is given as follows:

Initialization Phase
REPEAT
Employed Bees Phase
Onlooker Bees Phase
Scout Bees Phase
Memorize the best solution achieved so far
UNTIL (Cycle = Maximum Cycle Number

The main detailed steps of the ABC algorithm implemented for this BACAP resolution are presented as follow :

Algorithme 2.1 – ABC

1. Initialize parameters.
2. Generate the initial population of solutions x_i randomly which contain *NS* solutions (number of sources), in our case, a solution is a feasible berth and crane assignment plan. {**initialization**}.
3. Evaluate the fitness function $f(x_i)$ of all solutions in the population, where $f(x_i)$ is $\frac{1}{1+Total\ Service(x_i)}$.
4. Keep the best solution x_{best} in the population. {**Memorize best solution**}.
5. Set cycle =1
6. **Repeat**
7. Generate a new neighborhood solution υ_i from the old solution x_i performing a little perturbation to the berth and crane plan. {**Employed bees**}.
8. Evaluate the fitness function $f(v_i)$ for all solutions in the population.
9. Keep the best solution between current and candidate solutions {**Greedy selection**}.
10. Calculate the probability P_i, for the solutions x_i, where $P_i = \frac{f_i}{\sum_{i=1}^{NS} f_i}$.
11. Generate the new solutions v_i (neighborhood) from the selected solutions depending on its Pi {**Onlooker bees**}.
12. Evaluate the fitness function fi for all solutions in the population.
13. Keep the best solution between current and candidate solutions {**Greedy selection**}.
14. Determine the abandoned solution if exist, replace it with a new randomly solution x_i {**Scout bee**}.

14: keep the best solution x_{best} found so far in the poupulation .
15: *cycle=cycle+1*
Until cycle ≤ cycle max

As stated earlier, the ABC is an iterative procedure to search improvements for initial feasible solutions to achieve a near-optimal solution towards the end of its research cycles. The starting point will, therefore be a population of feasible solutions generated randomly. Thereafter, in the Employed Bees phase, a neighbor is generated for each solution of the initial population via the neighborhood procedure. The neighbor that is better than its associated solution is retained, otherwise the old solution is kept. Before proceeding to the next generation, the Scout Bees phase is performed to replace solutions that have not been improved during the process . The iterations follow one another until the maximum number of cycles is reached.

The proposed framework of ABC applied to the BACAP resolution is presented in Figure 2.5.

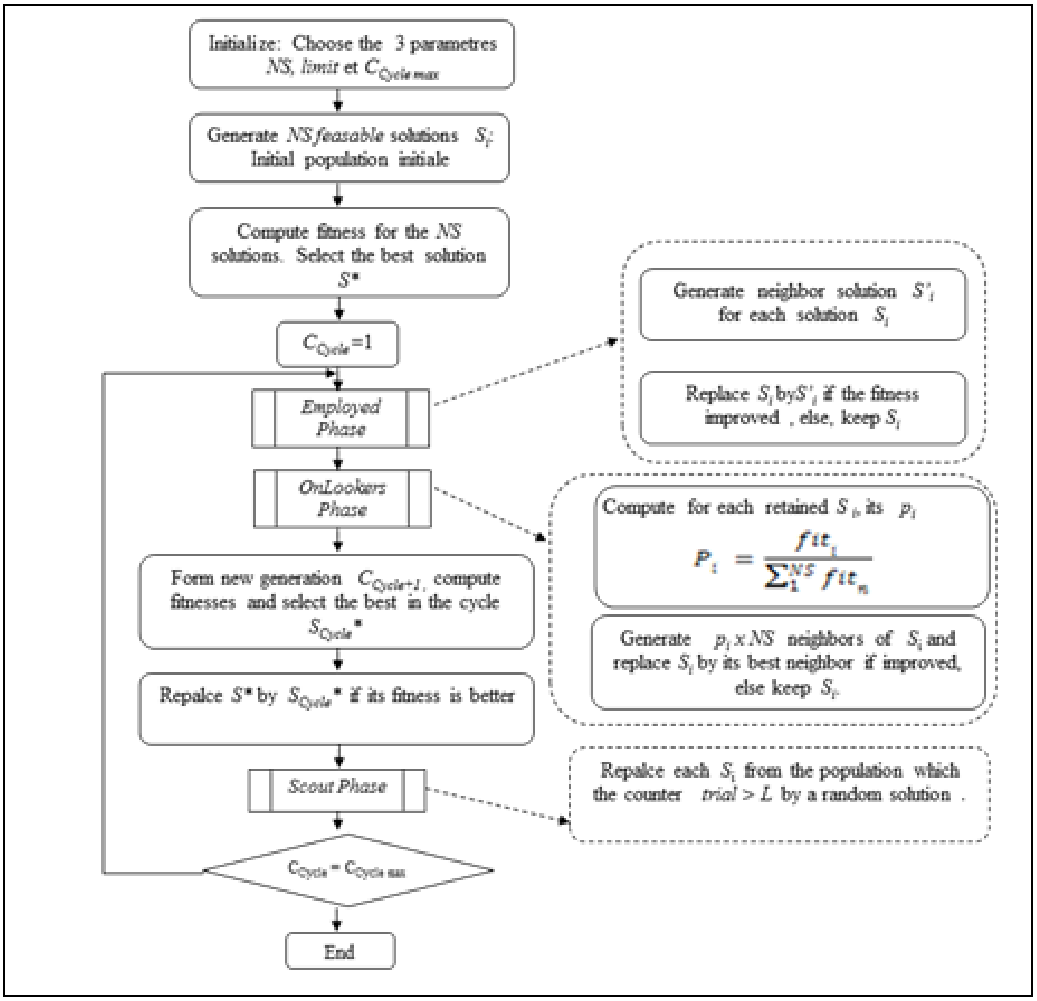

Figure 2.5 – Proposed framework for ABC

2.6 Experiments and Results

The *BACAP-TIA* has been implemented in Matlab language and the experiments have been carried on a PC with an Intel Pentium 2.2 GHz CPU and 4G DRAM.

This section presents some computational experiences considering problems of different sizes that range from 12 up to 40 vessels in order to reach performance conclusion regarding the instances. Parameters setting for ABC have been also changed to observe the impact of the change on the results.

The experiments concern a terminal with 7 berths, 3 of which (number 1, 6 and 7) are for container ships and 4 (number 2, 3, 4 and 5) are intended for RoRo. The number of cranes available at the terminal is 5.

Crane speed is set to 20 containers/hour and trailer charging/discharging speed is set to 30 trailers/hour. The two interference coefficients $coef_1$ and $coef_2$ are set respectively to 1.5 and 1.2.

2.6.1 Initial solution

An instance of 12 ships presenting the Input of the construction heuristic is shown in table 2.2. The Output represented by the Gantt 6 in figure shows the berth and crane assignment.

Tableau 2.2 – 12 ships Data example

	type	Arrival	Depature	Containers	trailers
1	Container-Ship	30	--	350	
2	Container-Ship	20	--	350	
3	Container-Ship	9	--	320	
4	Container-Ship	8	--	250	
5	Container-Ship	10	--	600	
6	Container-Ship	12	--	560	
7	RoRo	9	24	70	130
8	Roro	14	24	20	240
9	RoRo	30	68	55	254
10	RoRo	46	68	60	200
11	RoRo	32	72	100	180
12	RoRo	16	35	80	100

In Figure 2.6, an initial feasible solution is presented showing the berth and crane assignment. The colored rectangles represent ships (container ships and RoRo). The number of cranes assigned to each vessel is presented inside the rectangle (ie: ship 2 (dark blue), assigned to the first berth in the third service order has 2 cranes (C=2). The RoRo are assigned to their dedicated berths, (2, 3, 4 or 5), container ships could only be berthed at quays 1, 6 or 7. The rectangle length represents the time spent by the ship in the terminal; it is simply the handling time.

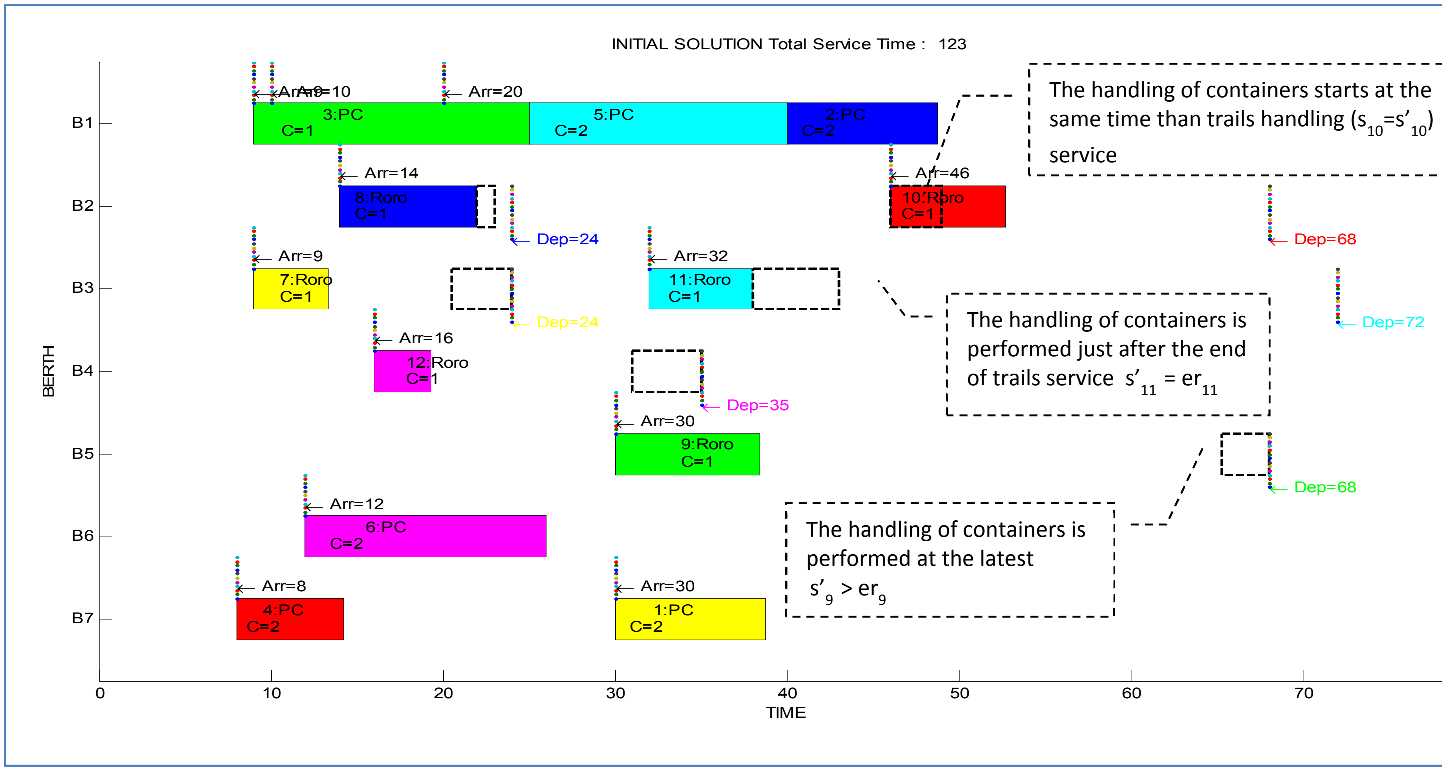

Figure 2.6 – Gantt chart for an initial solution of a 15 ships' instance

In case of RoRo, the coloured rectangle is the duration of handling trailers and the dotted lines represent the containers' handling service performed by 1 crane. This container handling could be at the berthing time (case of ships 8,and 10), and in that case, the two loading services (containers and trailers) are overlopped. Other cases are also possible, the containers' handling is performed at the end of trailers handling (ship 11) or could be performed at the latest but before the departure time (ships 7, 12, 9), in these cases, the two services are performed sequentially one after the other.

For this initial solution, the total service time is 123 hours. The Gantt shows that the ships 5 and 2 wait before berthing and, for this example, there is no delay, because no one of the RoRo leaves after its due time.

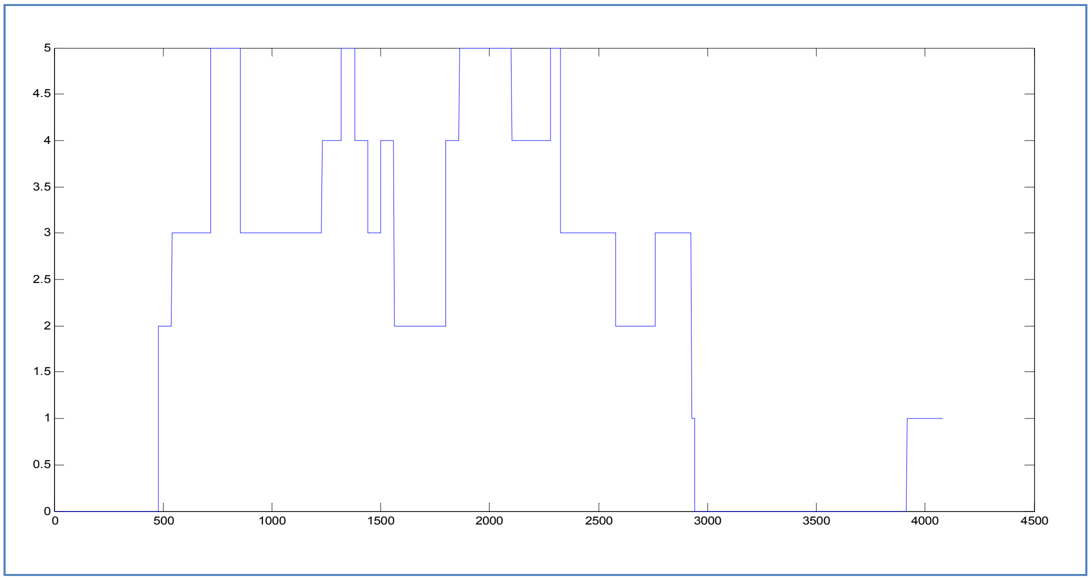

Figure 2.7 – Cranes Use during the time horizon

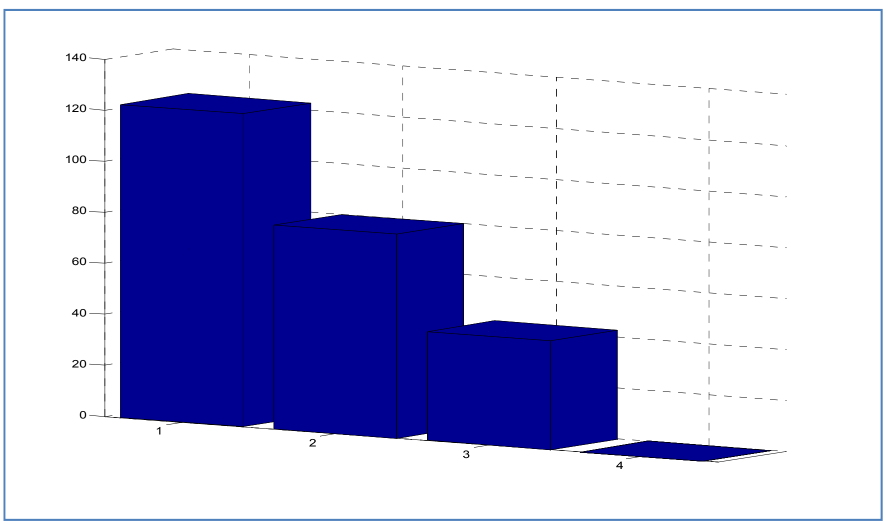

Figure 2.8 – Distribution of the different times of the solution

(1: Total Service time, 2: handling time, 3: waiting time,4:Delay)

2.6.2 Neighbourhood

To generate the neighbor, some perturbations to the solution are performed, i.e. in figure 2.10, the ship 11 affected initially (figure 2.9) to the berth 5 is changed to the berth 2. This change is performed with respect to the release time constraints (arrival time). Other changes concerning the handling starting for RoRo are proposed, (i.e. instead the container service starts at the arrival of the ship, it starts after the trailers handling. The shifting of the start of handling is done to satisfy the constraint of using up to the maximum number of cranes available. For that example, the neighbor solution is presenting a better result as total service time, it decreases to 139.43 hours.

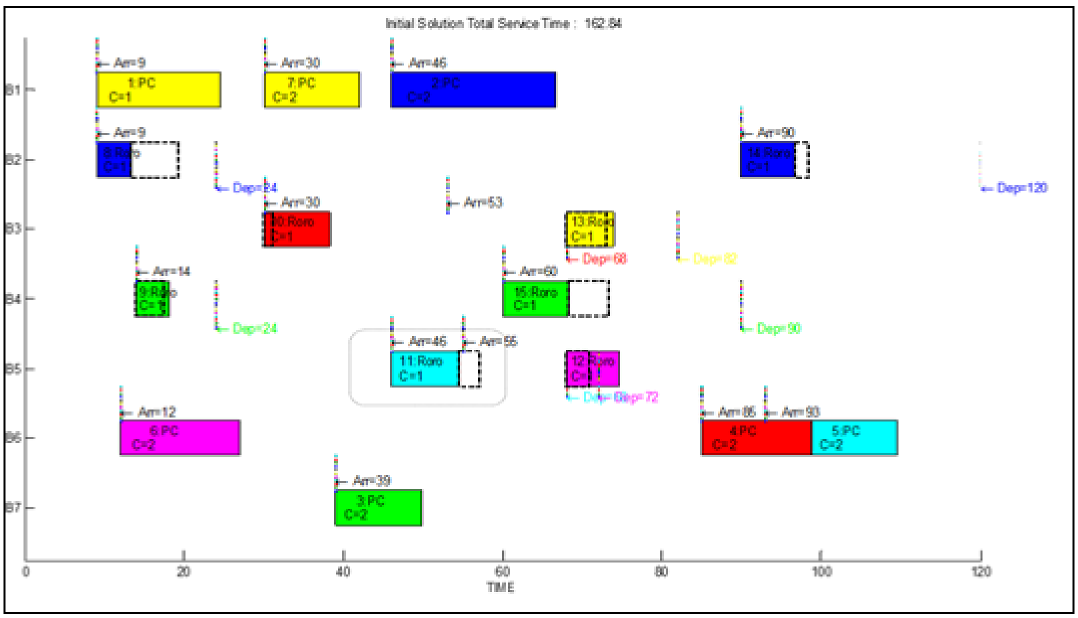

Figure 2.9 – Initial Solution before perturbations

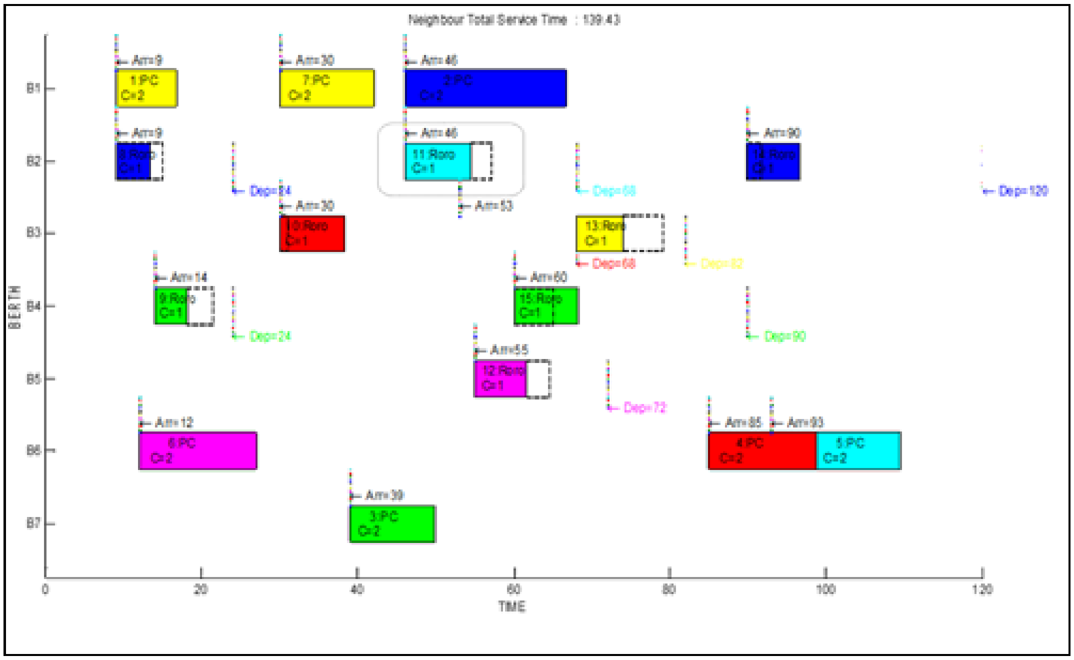

Figure 2.10 Neighbor Solution

2.6.3 DATA generation for experiments

To simulate the simultaneous berth and crane assignment process for different time horizons, ship data for experiments are randomly generated. Arrivals for the two types of ships, the departure of RoRo, capacities for containers and trails are generated from discrete uniform distributions.

2.6.4 ABC optimal solution search process

Once parameters set such as *NS, limit and* $Cycle_{max}$ are defined, 5 experiments are deployed for each combination. In each experiment, 10 simulations are run. Table 2.3 present results for an example of 10 run simulations.

Tableau 2.3 – 25 ships Example Results

25 ships ,Time Horizon 150h NS=50,limit=250 , Cycle $_{max}$=300			Decrease %
Simulations	Total Service Time for Initial solution (hour)	Total Service Time for Final solution (hour)	
1	223.00	196.80	13.31
2	230.81	199.80	15.52
3	227.68	194.50	17.06
4	227.96	194.50	17.20
5	217.76	194.50	11.96
6	230.98	196.50	17.54
7	221.67	196.50	12.81
8	227.64	195.50	16.44
9	*232.90*	*199.82*	16.55
10	223.81	195.50	14.48
	Standard deviation 4.75	Standard deviation 1.99	

Besides the time in hours, for the 10 runs of the same instance, the standard deviation is calculated to examine steadiness and accuracy of solutions, and effect of the initialization parameters on it. The rate of decrease for each run is also observed.

Figures 2.12 and 2.13 present respectively Gantt charts for the initial and final solutions for the ninth run of this test example. We can observe that, after 300 neighborhood cycles, we observe that the chart Gantt has completely changed to show the final results of berth and cranes' assignments . Convergence of the ABC is illustrated in Fig 2.11, for this example, the search process reached the final and better solution after almost 100 cycles, it is maintained the same for the next 200 cycles.

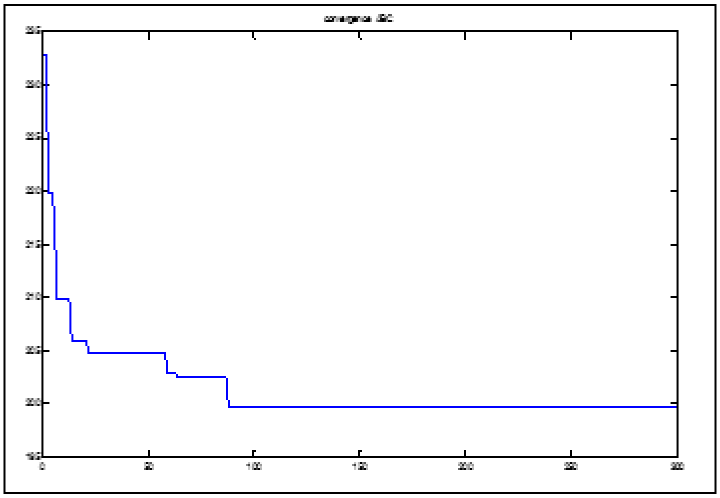

Figure 2.11 – ABC convergence

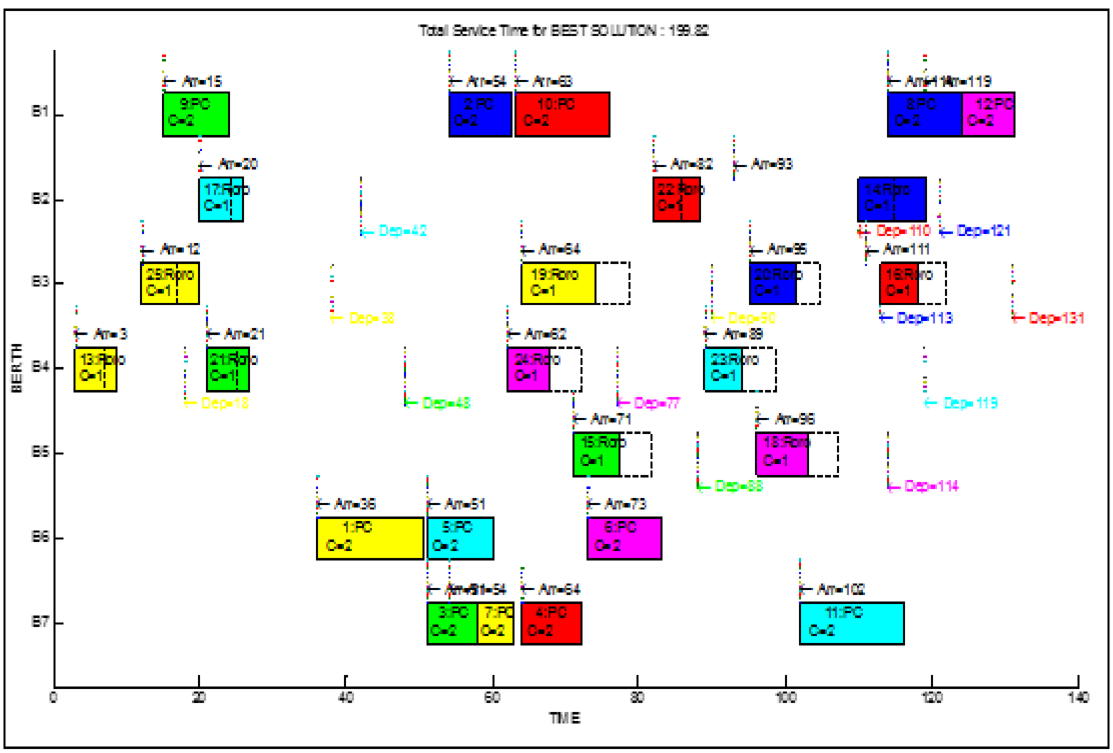

Figure. 2.12 – Gantt chart for an initial solution of 25 ships' example

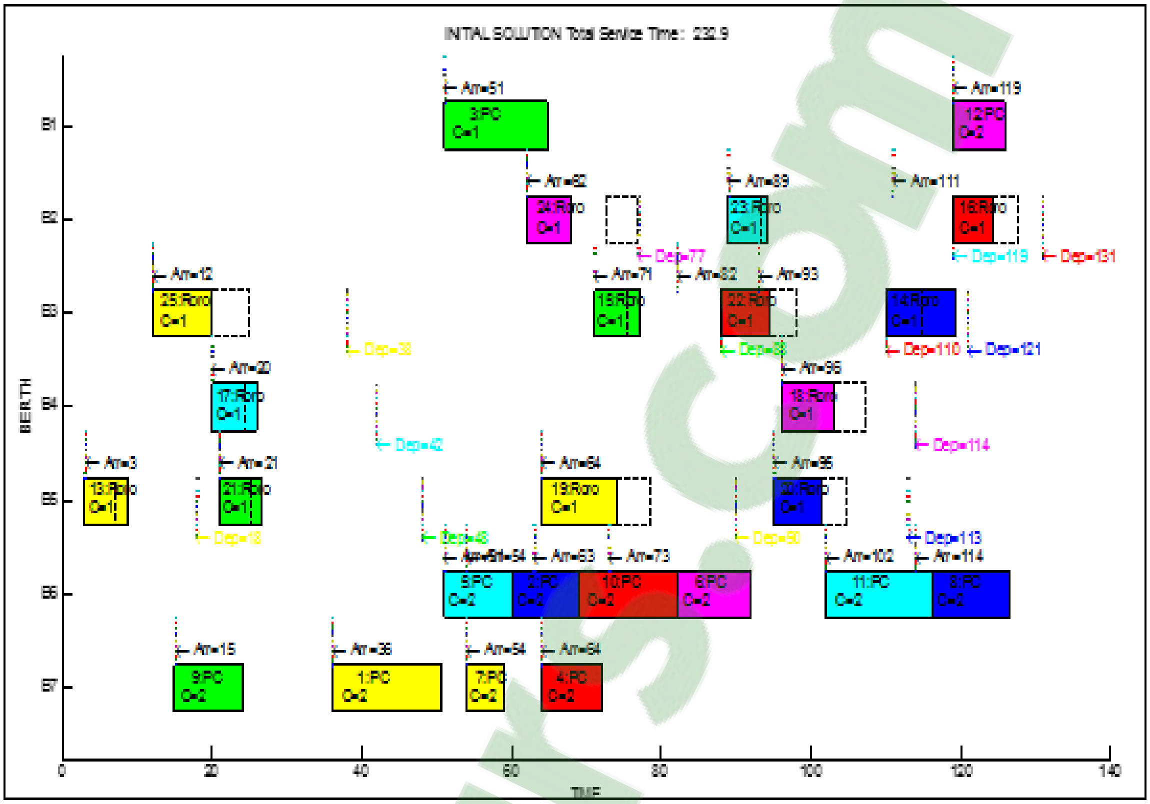

Figure. 2.13 – Gantt chart for the final solution of 25 ships' example

2.7 Results and Discussions

In Table 3 and to examine the performance of the BACAP-*TIA* model for large instances we randomly generate additional test instances by keeping the number of berths and the number of available cranes at the same value as before. We vary the number of vessels from 12 to 40 with varying tuning ABC parameters and sometimes time horizon span by increasing number of vessels (for large instance between 150h (6 days) and 190h (8 days)).

Tableau 2.4 – Results for different size problems

Problem Size	Experiments (10 simulations each)	Tuning ABC parameters (NS, *Limit*, $Cycle_{max}$)	Average Running time	Standard deviation for the initial solution	Standard deviation for the final solution	Max decrease (%)
12 ships 6 cont-ship 6RoRo 100h	1 2 3 4 5	10,15,20	1s	0.66 0 0 0.11 0	0 0 0 0 0	6 0 0 2 0
12 ships 6 cont-ship 6RoRo 100h	1 2 3 4 5	50,150,200	10s	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
15 ships 8 cont-ship 7RoRo 100h	1 2 3 4 5	10,15,20	1.4s	2.52 0 4.7327 1.35 0	0 0 0.4830 0 0	8 0 10 3.5 0
15 ships 8 cont-ship 7RoRo 100h	1 2 3 4 5	50,150,200	10s	0 0 0.64 0 0	0 0 0 0 0	0 0 6.8 0 0
20 ships 10 cont-ship 10RoRo 150h	1 2 3 4 5	20,50,100	3.2s	5.12 3 4 7.5 1.5	0 0 1.4 0 0	17.3 5.3 15 18 4
20 ships 8 cont-ship 12 RoRo 150h	1 2 3 4 5	50,250,300	14 s	6.25 5.4 1.9 6.14 3.7	0 1.05 0 0 0	14.7 15.7 4.7 17.8 21
20 ships 10 cont-ship 10RoRo	1 2 3 4 5	50,250,500	23s	3 2.26 1.2 3.5 0.5	0 0 0 0 0	14.5 3.8 2 5.8 1.6
25 ships 12 cont-ship 13 RoRo 150h	1 2 3 4 5	20,250,300	7s	8.7 6.15 7.9 9.8 4.5	2.9 0 3.8 3.5 0	18.25 19.8 19.3 40.5 20.3

Tableau 2.4 (suite)

Problem Size	Experiments (10 simulations each)	Tuning ABC parameters (NS, *Limit*, $Cycle_{max}$)	Average Running time	Standard deviation for the initial solution	Standard deviation for the final solution	Max decrease (%)
25 ships 12 cont-ship 13 RoRo 150h	1 2 3 4 5	75,250,300	22.5	5.28 6 5.24 3.7 6.95	0.44 1.5 0.13 0 0	27.23 15.9 13.4 12.6 11.93
25 ships 12 cont-ship 13 RoRo 150h	1 2 3 4 5	75,50,300	28s	8.99 4.35 4.96 5.78 6.10	3.17 1.22 1.32 2.41 1.79	24.44 17.19 20.3 20.94 26.02
25 ships 12 cont-ship 13 RoRo 150h	1 2 3 4 5	50,250,500	25 s	8.17 3.6 6.9 5.87 8.3	1.35 0 0.15 0 1.8	23.73 19.6 16.1 14.4 27.1
30 ships 15 cont-ship 15 RoRo 150h	1 2 3 4 5	75,250,300	25s	6.99 6.52 5.46 8.79 7.39	0.31 2.86 2.07 4.54 2.95	9.82 17.82 19.18 28.36 26
30 ships 15 cont-ship 15 RoRo 150h	1 2 3 4 5	75,250,300	25s	6.99 6.52 5.46 8.79 7.39	0.31 2.86 2.07 4.54 2.95	9.82 17.82 19.18 28.36 26
30 ships 15 cont-ship 15 RoRo 170h	1 2 3 4 5	75,250,300	23.5s	6.8 6.25 3.86 6.50 5.29	0.94 0 0 0.05 0.31	20.78 12.18 10 19.65 18.93
30 ships 15 cont-ship 15 RoRo 150h	1 2 3 4 5	75,250,500	40s	8.37 10 6.49 7.79 6.84	0 0 0 0.51 0.51	20.62 15 17.30 17.2 21.34

Tableau 2.4 (suite)

Tableau 2.4 Suite Problem Size	Experiments (10 simulations each)	Tuning ABC parameters (NS, *Limit*, $Cycle_{max}$)	Average Running time	Standard deviation for the initial solution	Standard deviation for the final solution	Max decrease (%)
35 ships 15 cont-ship 20 RoRo 170 h	1 2 3 4 5	75,250,500	45s	2.28 7.21 6.43 5.92 4.14	2.57 0.96 0.70 0.06 2.28	15.23 14.9 10.73 11.25 19.27
35 ships 15 cont-ship 20 RoRo 170 h	1 2 3 4 5	100,500,1000	100s ~ 1.6 min	5.65 2.65 4.56 2.98 4	0 0.05 0 0 0	21.73 11.89 13.13 10 12.45
40 ships 20 cont-ship 20 RoRo 190h	1 2 3 4 5	50,250,300	19s	11.13 8.70 6.28 8.29 13.29	1.7 2.82 2.40 3.85 2.11	18 19.42 20.87 27.74 25.9
40 ships 20 cont-ship 20 RoRo 190h	1 2 3 4 5	100,500,1000	80s	4.76 5.23 7.12 2.4 3.3	0 0.07 0.12 1.06 0.89	12.34 16.78 21.56 10.10 12.34
35 ships 15 cont-ship 20 RoRo 170 h	1 2 3 4 5	100,1500,2500	245 s ~ 4 min	8.63 5.5699 6.2 7.83 4.65	0 0 0 0.07 0	17.83 11.54 21.17 18 15.67

To summarize the table of results, each experience is simulated by 5 x 10 runs and the experimental simulation having the minimum standard deviation for the final solution gives the near global optimum result. In Table 2.4, several experimental gave zero as standard deviations , this means that the final solutions for the 10 run simulations for the same instance converge towards the same result, which proves the effectiveness of the algorithm.

Initial solutions for each experimental simulations are not very scattered, that is due to the number of sources *NS* in the ABC algorithm since the best among these population of first feasible solutions .

As can be observed, even the largest instance with 40 vessels can be solved to global optimality in nearly 5 minutes, which is very fast and proves that ABC time performance is somewhat satisfactory.

The increase of $Cycle_{max}$ is performed when we notice in the convergence curve that there is still a minimization of the total service time approaching current $Cycle_{max}$, (ie, Fig.2.14). This might suggest that it is possible to achieve better results by increasing the number of iterations.

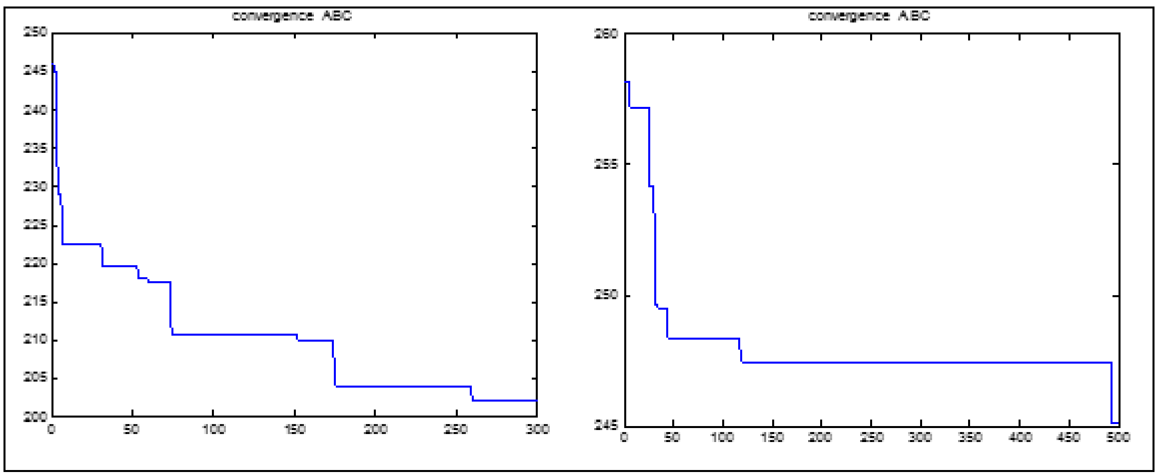

Figure 2.14 – ABC convergence for 300 and 500 Cycles

Furthermore, it is interesting to note that the parameter *NS* has an effect on Initial solution which is very important for the search process , while the parameter $Cycle_{max}$ has an impact on the final solution. *Limit* is the parameter that allows the algorithm to escape local- optima due to the discovery of other zones of feasible solutions. These findings are based on the observations of results of several simulations conducted apart from the ones listed in Table 2.4. For small size instances (12 and 15 ships), the solving approach is able to reach very fast the near global-optimum, whereas it takes much more time for large ones, especially when the tuning parameters are increased.

2.8 Conclusion

In this study, a new non linear model for a planning problem inspired from a real case container terminal is presented. This paper addresses the efficient scheduling of simultaneous berth and time invariant quay cranes at a special container terminal which accommodates two types of vessels.

Since these kind of problems are known for their complexity of resolution (called NP hard), we propose a solving approach based on an Ant Bee Colony meta-heuristic.

The numerical results show that the proposed solving approach is a promising algorithm and can obtain the optimal solution or approximate optimal solution for our *BACAP-TIA* in a very fast time simulation. The proposed Artificial Bee Colony algorithm is found very effective in solving small to large sized problems.

Actually, the proposed algorithm easily found near optimal solutions based on the convergence of the results of the various instances simulations.

Further research, in progress, concentrates on developing a model taking into consideration additional practical factors such as variable in time crane assignment and comparing other solving apprach.

Acknowledgement

The author would like to acknowledge the generous assistance and valuable information provided by the employees of the anonymous Container Terminal .

Conflict of interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

CHAPITRE 3

SIMULTANEOUS DEDICATED BERTH ALLOCATION AND CRANE VARIABLE-IN-TIME ASSIGNMENT PROBLEM IN A SPECIAL CONTAINER TERMINAL

El Asli Neila[1], Dao Thien-My[1], Bouchriha Hanen[2]

[1]Mechanical and Industrial Engineering, École de Technologie Supérieure (ETS)
Montreal, Canada,

[2]Industrial Engineering, National Engineering School of Tunis (ENIT),

This chapter has been submitted to « Flexible Services and Manufacturing Journal », Springer

3.1 Abstract

Berth allocation and crane assignment problem is an interesting integration of two major problems in the container terminals ' logistic. In this paper we present an application of these two simultaneous problems in a very particular and real case where the terminal is treating two different types of vessels simultaneously; container ships and Roll-on Roll-off (Ro-Ro) ships. A new mixed Integer nonlinear model (MIP) is presented. An event-based heuristic for the initial solution construction is proposed. The search optimal solution process is performed by the Artificial Bee Colony (population) meta-heuristic and an Extended Great Deluge algorithm (local search). Results provided by these two meta-heuristics are then compared and discussed.

Keywords: Terminal Operations, Berth Allocation, Crane Assignment, Container Vessel, Ro-Ro Vessels, Event-based heuristic, Variable –in-time assignment, Artificial Bee Colony meta-heuristic, Extended Great Deluge Algorithm

3.2 Introduction

Container terminals are the areas where containers are transported from one point to another using different handling equipment. Such terminals are continually growing in importance as maritime transport faces the challenge of using new technologies to build larger and larger ships. Moreover, transport frequency is only rising as commercial exchanges are developed to meet economic growth. To be able to compete within this environment, container terminals must be managed efficiently. To that end, managers must concentrate on the most critical resource for determining container terminal capacity: Berth. An alternative approach to increasing Berth capacity involves improving its productivity through its efficient use (Park and Kim, 2003). One of the components of such efficient utilization is a focus on quay cranes, which are the main equipment used to move containers at terminals.

More and more studies are being dedicated to the examination of container terminals and efficient operations which improve their productivity. We shall focus, then, our study on the most attractive integrated seaside problem, which increasingly attracts the interest of researchers: the simultaneous berth allocation and crane assignment problem, known as BACAP. The Berth allocation problem is a well-studied between NP-hard combinatorial problems (Lim 98) in the terminal planning field. It consists in assigning incoming vessels to berthing positions. The crane assignment problem deals with assigning quay cranes to each vessel for handling the containers on board. These problems integrated together are over complex, and studies in the literature are constantly approaching them for modelization in a realistic way and solving them by efficient methods. In this work, both issues are mentioned in the case of a special terminal. A special container terminal reality has inspired the formulation of a new nonlinear mathematical model, with assumptions for simplicity, and modern approaches to the operation research are there for solving.

After a brief literature review focused on the integrated berth and crane assignment problem and specifically, the variable-in-time assignment, a new non-linear formulation is proposed and discussed. Section 3 is dedicated to present the two meta-heuristic-based solving approaches, the Artificial Bee Colony, and the Extended Great Deluge Algorithm, and precisely the event-based construction heuristic for the initial solution. The last section is reserved to present the results

3.3 Literature review

The integrated Berth and Crane Assignment problem has been widely discussed in the literature and in the two surveys published in 2010 and 2015 by Bierwierth and Meisel remain very good references for the reader wishing to have a complete vision in relation to these seaside terminal operation problems ranging from the physical description of these problems to the classification of the studies dealing with their modelization and resolution, and finally trends in the field. In this paper, we attempt to review some of the studies focusing on simultaneous berth and crane assignment problems since the follow-up survey of Bierwieth and Meisel (2015) and distinguish between those with invariant-in-time assignment from those with variable-in-time assignment. According to Meisel in his book, (2009), the assignment of cranes to a single vessel is called a QC-to-Vessel assignment. Basically, a terminal management chooses one out of two strategies for generating QC-to-Vessel assignments:

- The number of cranes assigned to a vessel is kept fixed throughout the service process, which is referred to as a time-invariant assignment (see Vessels 1, 4,and 5 in Fig. 3.1.b).
- In a variable-in-time assignment, the number of cranes assigned to a vessel can change during the handling time (Vessels 2 and 3 in Fig. 1.b). (Meisel,2009).

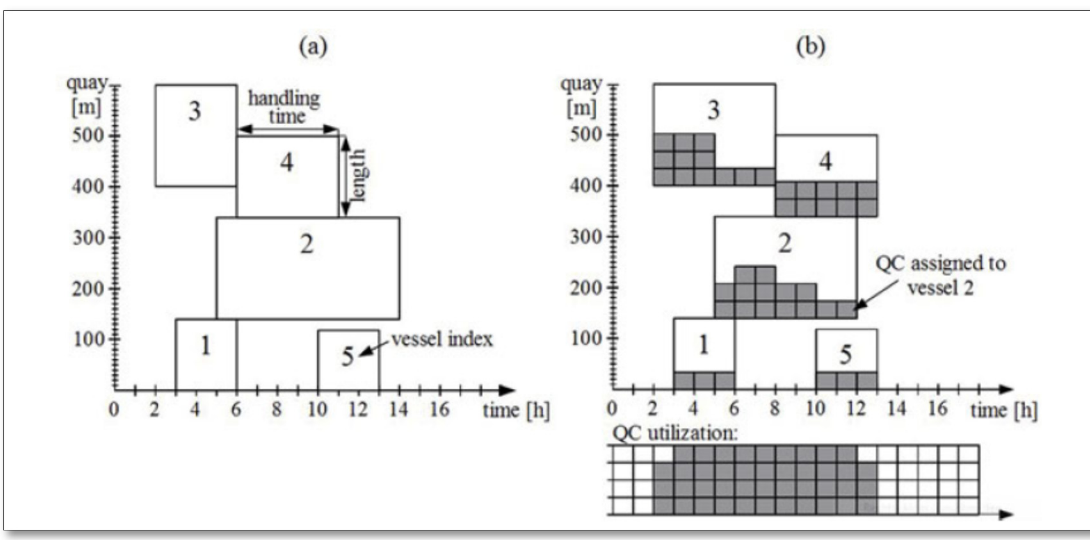

Figure 3.1 – Service plan (a) complemented by a crane assignment (b)

More and more researches focus on the second strategy, which is more realistic and needs more innovation to formulate models and find solutions. In fact, since 2015, the major part of studies dealing with the two integrated planning problems has been presenting variable-in-time strategy for the crane assignment. Table 1 lists some of these studies' references. New models and solving approach were proposed to capture more practical requirements for this integrated issue.

Tableau 3.1 – Recent integrated studies studies

Crane Assignment Strategy	Reference
Time Invariable Assignment	Han and al; Lalla-Ruiz and al.(2015); Agra and al. (2016)
Variable in Time Assignment	Hu (2015), Xiao and Hu(2015); Iris and al. (2015); Ji and al.(2015); Turkogullari and al. (2016); Salhi and al. (2017); Karam and al. (2016); Hsu (2016)

Among the invariant-time-strategy assignment studied, Agra and al. (2016) formulated a new mixed integer programming resulting from a discretization of the time and space variables. The objective is to minimise the completion time. They considered a heterogeneous set of cranes. The resolution method was based on a branch and cut algorithm after finding an upper bound by a rolling horizon heuristic.

Han and al. (2015) proposed two phase-model for the berth allocation and crane assignment problem, in which multi-objective programming was formulated. The berth allocation phase aimed to minimize stay time and increased cost caused by deviation berthing and the crane assignment phase minimized both range of cranes used and movement of cranes. The solving approach was based on a particle swarm optimization.

Lalla-Ruiz and al. (2015) presented for the first time Migrating Birds Optimization-based approach for addressing the two essential seaside problems; Berth allocation and Crane scheduling.

As stated earlier, the variant-in-time assignment strategy kept more the attention these last few years. Karam and al. (2016), proposed a new functional integration approach for the berth allocation, the quay crane assignment and the specific quay crane assignment, using a feedback loop structure approach. Their objectives were the minimization weighted sum holding cost and the minimization of the average holding time per vessel.

He (2016) developed an bi-objective mathematical model for berth and carne assignment problem aiming to minimize total departure delay and total energy consumption of all vessels. Integrated simulation and optimization method was proposed for resolution using the Simulated annealing algorithm as a local search.

Hu (2015) developed a rolling-horizon heuristic algorithm for solving a continuous berth allocation problem considering periodic balancing utilization of cranes. based on hybrid heuristics of mathematical programming, neighborhood search, and parallel computing. While most studies were seeking solutions under cost minimization, the study has exploited the maximization of crane utilization in the duration of each work shift.

Turkogullari and al. (2016) formulated by a mixed integer linear program a simultaneous berth allocation, quay crane assignment and scheduling problem. The authors proposed an efficient cutting plane algorithm based on a decomposition scheme in order to solve the problem efficiently based on the separate solution of the scheduling sub-problem at each step.

Salhi and al.(2017) integrated optimisation model combining three seaside planning problems, with the objective being to minimize the tardiness of vessels and reduce the cost of berthing. An implementation of the genetic algorithm is considered and the results are compared to those of Cplex.

Hsu (2016) proposed a hybrid particle swarm optimization with an event-based heuristic to solve a non-linear programming for the integrated berth and variable-in-time crane assignment.

In this study, our event-based heuristic is more enriched in terms of events compared to Hsu'(2016) one . The later studied only the berthing and leaving events, while we propose several other events discussed in the following.

3.4 Problem description and mathematical model

Our case of study is very specific as stated earlier; in fact, the terminal is accosting two types of ships with intended berths for each. Cranes are mobile and shared between the first type, which is feeder container ship, and the second, which is Roll-on Roll-off (RoRo) vessel, loaded with trailers and containers. Mobile cranes, usually dedicated to container ships, are used to handle the containers aboard these RoRo, whereas tare charged/discharged using internal terminal trucks as a horizontal charging.

Fig. 2 shows the layout of the berth area, which is partitioned into 6 discrete accosting zones. Berths 1, 6 and 7 are intended for the container ships and berths 2,3,4 and 5 are intended for RoRo.

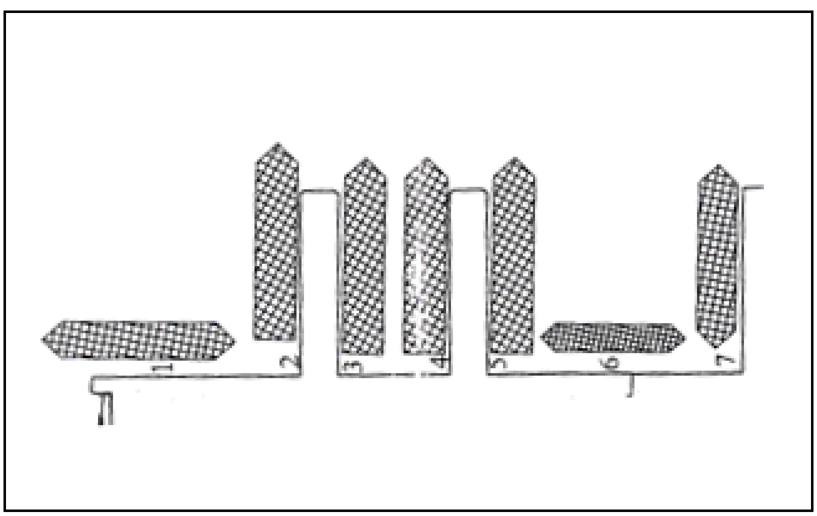

Figure 3.2 – Layout of the Berth area

Ro-Ro ships are vessels that are used to carry wheeled cargoes, such as cars, trucks, semi-trailers trucks, trailers and railroad cars that are driven on and off the ship having their own wheels or using a platform vehicle. Ro-Ro ships that accost at their intended berths are also often partially loaded also with containers. This leads to a probable use of handling resources such as mobile cranes used by container ships. This sharing of a resource leads to an operation productivity decrease.

Another distinguishing feature of this container terminal is that only RoRo are constrained by a time window (arrival and departure time) while container ships arriving at the port are not faced with an imposed due date. Indeed, they are feeders (a feeder is a small vessel which makes the pre and post container transport to ports with no stop), which arrives at the container terminal carrying, exclusively, the dedicated containers.
The incoming RoRo berthed at the terminal, remain at their positions throughout all the time window and leave only at their due dates.

Usually, managers at container terminals face three interrelated decisions in BACAP context: where and when the vessels should moor, and how many cranes should be assigned to each one.
The objective of the managers in that terminal would be minimizing the total time for these two types of ships, namely the waiting time, handling time and delay time for the Ro-Ro ships which occur if a Ro-Ro ship leaves after its due time.

For the current situation, the container handling is done systematically following the rule of first-come, first-served basis, in accordance with the dedate of departure of the Ro-Ro ships and by sharing the cranes. There is no exact procedure for sharing; it means that cranes moving between container ship and Ro-Ro can be done at any time during a work shift by interrupting the container ship handling if necessary since the Ro-Ro is a priority. As such, a better policy for planning and the assignment of the cranes is requested.

Finally, to sum up, managers at this container terminal have to decide where and when the vessels should moor, how many cranes should be assigned to each container ship, and exactly when a crane transfer should be done between a container ship and Ro-Ro when the latter is requesting a crane to achieve handling operations.

This goal has inspired the following new formulation of the nonlinear mathematical model for Variable in-Time Assignment based on some assumptions to simplify the complexity of the terminal reality.

For practical reasons, some realities are not considered in this model such as the interruption of container handling when cranes are needed for the RoRo ships. Our tasks are assumed to be non-preemptive in this formulation.
In the real context, the terminal is working three 6 hour-shifts with 6 hours- interruption for break. In this model,time horizon is assumed to be without any breaks. Time for crane transfer is ignored too.

Inputs:

$i = 1, \dots, I\ set\ of\ vessels$;

$I = \{I_{PC}\ \&\ I_{RoRo}\} I_{PC}\ and\ I_{RoRo} are\ Containers\ Ships\ \&\ RoRo\ ships$

$j = 1, \dots, J\ set\ of\ discrete\ berths$

$;\ j = \{1,6,7\}\ \ for\ Containers\ Ships\ j = \{2,3,4,5\}\ \ for\ RoRo\ ships$

$k = 1, \dots, K; set\ of\ ship\ services\ at\ the\ same\ berth$

$m_i = 1, \dots, M_i; set\ of\ ship\ events\ during\ service\ ; i \epsilon I$

$a_i = ship\ arrival\ time\ ; i \in I$

$d_i = ship\ departure\ time\ ; i \in I_{RoRo}$

$C_{m,i} = Ship\ Conaitner\ Capacity\ \ ; i \in I$

$R_i = Ship\ trailers\ Capacity\ \ ; i \in\ I_{RoRo}$

$v = Crane\ Speed\ (Cont/hour)$

$v_R = Truck\ Speed\ (trailer/hour)$

$H = number\ of\ available\ mobile\ Crane\ )$

$$w_{m,i} \begin{cases} 1{,}5 & if\ h_{m,i} = 2 \quad ;\ i \in I_{PC} \\ 1 & if\ h_{m,i} = 1 \end{cases}$$

$w_{m,i}$ *= cranes congestion coefficient*

$coef_2 =$

$Interference\ coefficient\ if\ simultaneous\ container\ and\ trailers\ handling$

$sm\ =\ security\ margin\ =\ 1.3$

$$b_1 = \begin{cases} 2 & if\ C_{m,i} \geq 300 \quad i \in I_{PC} \\ 1 & else \end{cases}$$

Decision Variables :

$$x_{ijk} = \begin{cases} 1\ if\ the\ ship\ i\ \ is\ served\ \ at\ berth\ j\ as\ the\ k^{th}\ ship \\ 0\ else \end{cases}$$

$h_{m,i} =\ integer :\ \{0,1,2\},$

$number\ of\ cranes\ affected\ to\ \ ship\ i\ at\ event\ , m_i\ ;\ i\ \in I$

$s_i =\ Starting\ service\ time\ of\ ship\ i, i\ \in I$

$s'_i = starting\ trailers\ hanling\ time\ of\ ship\ i, i\ \in\ I_{RoRo}$

$f_i =\ Finishing\ service\ time\ of\ ship\ i, i\ \in I$

$e_i = Ending\ container\ handling\ time\ of\ ship\ i, i\ \in I$

$er_i = Ending\ trailers\ handling\ time\ of\ ship\ i, i\ \in I_{RoRo}$

$z_{it\,=}\quad (d_i - \frac{c_i}{v}).sm\ ;\ start\ of\ danger\ zone;$

$latest\ starting\ handling\ forRoRo$

$$y_t = \begin{cases} 1 & if\ an\ event\ happens \quad (event\ 1, or\ event\ 2\ or\ event\ 3) \\ 0 & else \end{cases}$$

Objective fonction :

$$Min\ Z = \sum_i \sum_j \sum_k T_{handling}\, x_{ijk} + \sum_i \sum_j \sum_k (s_i - a_i)\, x_{ijk} + \sum_i \sum_j \sum_k (f_i - d_i)\, x_{ijk} \tag{3.1}$$

Subject to :

$$\sum_j \sum_k x_{ijk} = 1 \quad \forall\ i \epsilon I \tag{3.2}$$

(3.3)

$$\sum_i x_{ijk} \leq 1 \qquad \forall j \, \forall k$$

$$s_i \geq a_i \qquad \forall i \epsilon I \tag{3.45}$$

$$\sum_i s_i \; x_{ijk} \quad \geq \quad \sum_l \left(s_l + \left(T_{handling}\right)_l\right) x_{ij(k-1)} \qquad \forall j \in \{1,6,7\}; \; \forall k \tag{3.5}$$

$$\sum_i s_i \; x_{ijk} \quad \geq \quad \sum_l max(f_l; \; d_l) \, x_{lj(k-1)} \qquad \forall j \in \{2,3,4,5\}; \forall k \tag{3.6}$$

$$h_{m,i} \quad \leq \; b_1 \qquad \forall i \epsilon I_{PC} \tag{3.7}$$

$$h_{m,i} \quad \leq \; b_2 \qquad \forall \; i \epsilon I_{RoRo} \tag{3.8}$$

$$h_{m,i} \;\; \leq \;\; H \quad \forall i \in I; \forall m \tag{3.9}$$

$$C_{m+1,i} \;\; = \quad C_{m,i} - \left[(t_{m+1} - \; t_m). \;\; v. h_{m,i}\right] \qquad \forall i \epsilon I_{RoRo} \tag{3.10}$$

$$T_{(handling)_i} = \sum_{m_i} (C_{m+1,i} \; - \; C_{m,i} \;) \,. \frac{w_{m,i}}{v. h_{m,i}} \quad ; \; \forall i \epsilon I_{PC} \tag{3.11}$$

$$T_{(handling)_i} = \frac{C_i}{v} \; .coef_2 \; ; \; \forall i \epsilon I_{RoRo} \tag{3.12}$$

$$er_i \; = \; s_i + \frac{R_i}{v_R} \;\; \forall i \in I_{RoRo} \tag{3.13}$$

$$e_i = \begin{cases} s_i + \; T_{(handling)_i} \;\; ; \; i \in I_{PC} \\ \quad s'_i + \dfrac{C_i}{v} \;\; ; \; i \in I_{RoRo} \end{cases} \tag{3.14}$$

$$th_i \; = \; s_i + (C_i \; - \frac{200}{h_i. v}) \, \forall i \in I_{PC} \tag{3.15}$$

$$f_i \; = \; e_i \;\; ; i \in I_{PC} \tag{3.16}$$

$$f_i = max(er_i \; ; \; e_i \,) \quad ; i \in I_{RoRo} \tag{3.17}$$

$$max(s_i \; ; \; s'_i\,) \;\le\; d_i \quad ; i \in I_{RoRo} \tag{3.18}$$

Before detailing the constraints, Figures. 3.3, 3.4 and 3.5 illustrate the main parameters of our model.

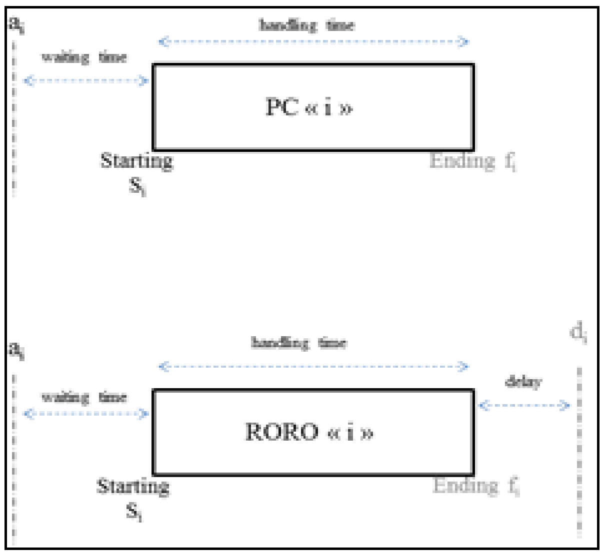

Figure 3.3 – The two types vessel parameters

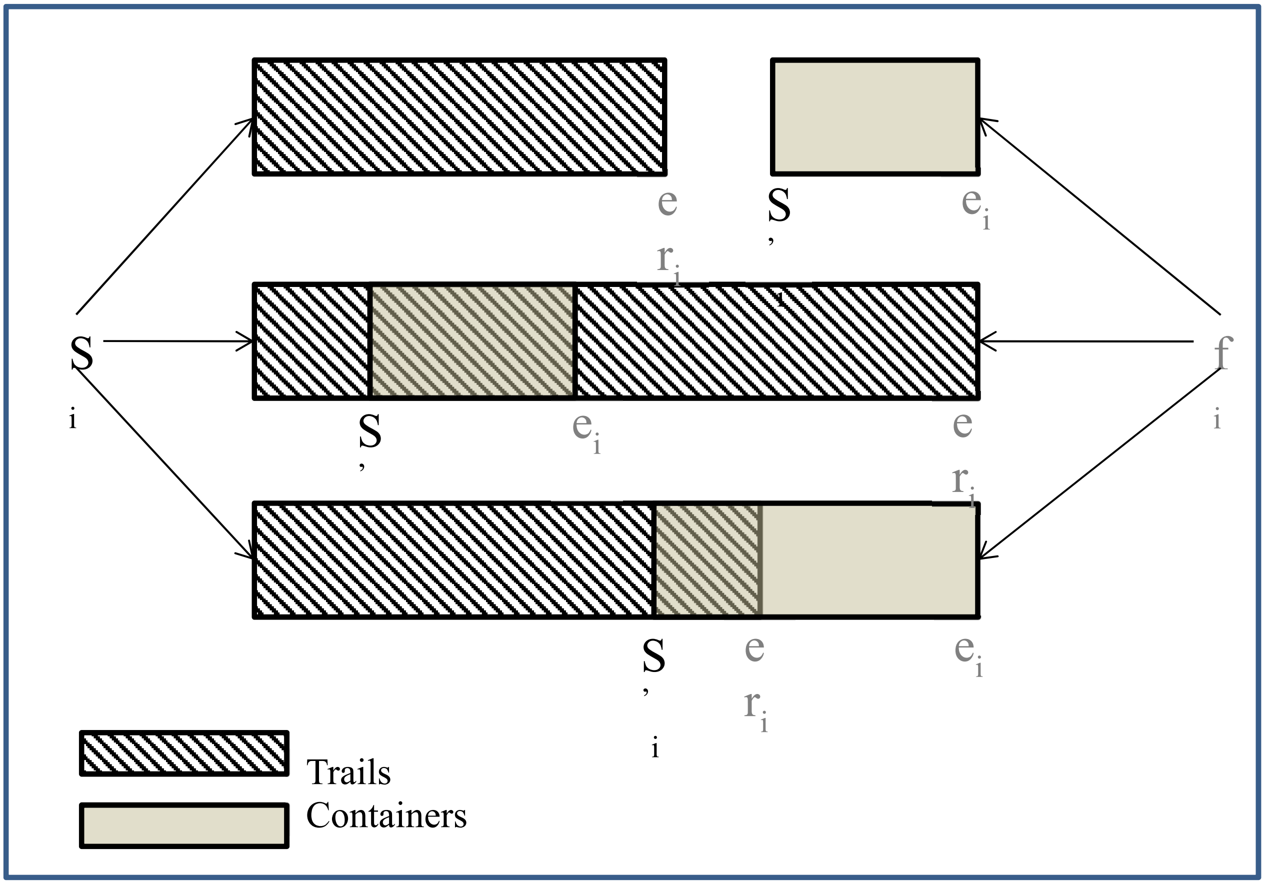

Figure 3.4 – Possible configurations for RoRo handling

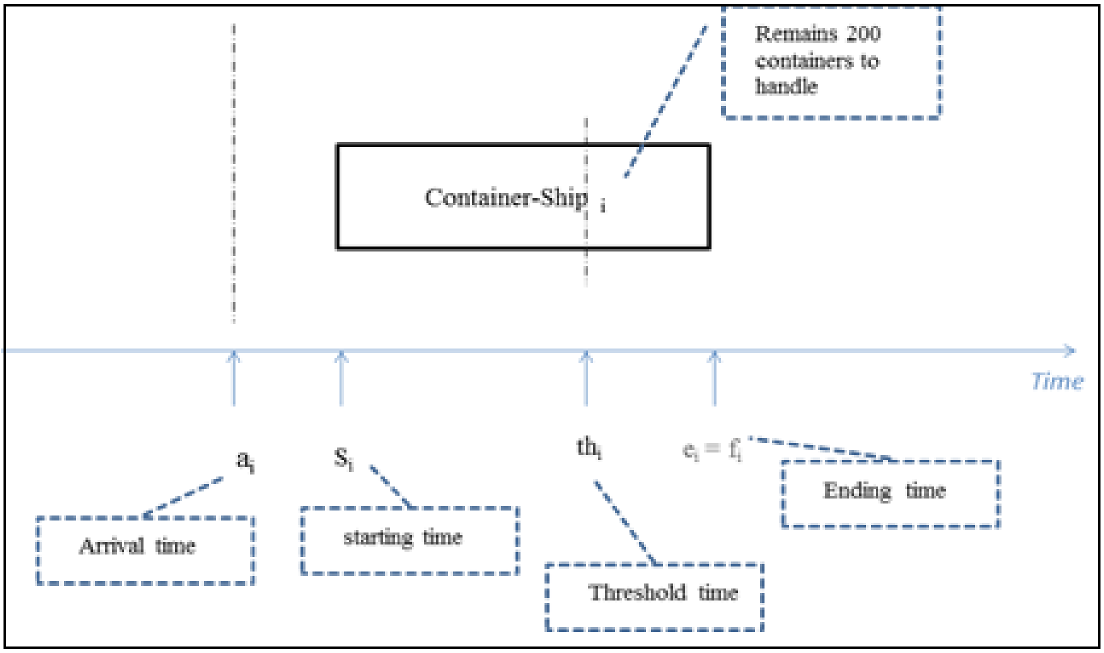

Figure 3.5 – Container ship

The objective function (3.1) minimizes as stated earlier, the summation of: 1- the waiting duration between the service starting moment and the arrival time , 2- the handling service duration for both types of vessels, and 3- the delay, in case it should happen (if the vessel leaves the terminal after its due time), for the RoRo vessels.

Constraint (3.2) forces that only one vessel can be served at each single berth at a time. Constraint (3.3) restricts every berth serves up to only one vessel at any time or unoccupied. Constraint (3.4) indicates that the start of the service only begins with or after ship arrival. Constraints (3.5) and (3.6) indicate respectively that, each container ship and RoRo ship cannot be served at any berth once its predecessor leaves. Constraints (3.7-3.8) define the maximum number of simultaneous cranes that could be assigned to containers vessel (2 maximum or 1 depending on charge) and RoRo-vessel (only 1). Constraint (3.9) ensures that assigned cranes do not exceed the available ones at any moment. Equation (3.10) calculates the number of containers to be loaded/unloaded remaining on board the container ship i at the event mi. Equations (3.11) and (3.12) formulates respectively the total container-handling times for container-ship and RoRO. Constraint (3.13) defines the ending time for horizontal loading/unloading (er_i) and constraint (3.14) defines the ending time for containers loading/unloading (e_i) for both ship types. Equation (3.15) defines the threshold time for container ship i. Constraint (3.16) and (3.17) define respectively the finishing times for the containers ships and RoRo. Finally, constraint (3.18) enforces RoRo starting time do not exceed its departure due time.

The mathematical model formulated above is a mixed integer nonlinear programming (MIP); which is not suitable for solving with conventional methods and commercial tools. However, the objective and constraints can be referred to develop heuristic approaches. In the next section, a proposed solving approach which combines event-based heuristic and metaheuristics is proposed to deal with these simultaneous planning problems. Fig. 3.6, illustrates the 3 steps of this solving approach. Details for each one are presented in the following section.

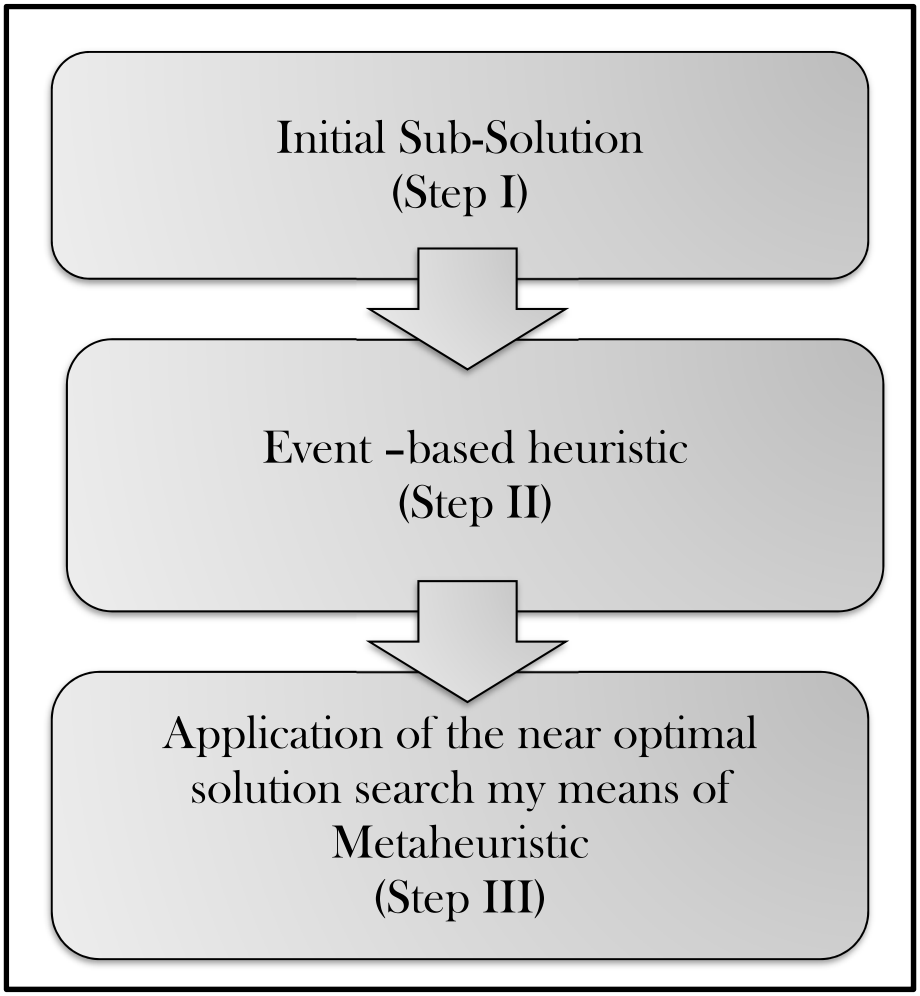

Figure 3.6 – Three Steps solving approcah

3.5 Solving approach for dedicated berths and crane variable in time assignement

3.5.1 Step 1: Initial sub-solution

The first stage for this solving approach consists in finding the first feasible sub-solution. The term sub-solution is used because the output of this stage is not a complete crane assignment for all incoming ships. Indeed, for this first step, a berth allocation is performed considering the intended berths, but, only, the container ships are served by a number of cranes. RoRo which are not supposed to start container handling at their starting service, have no cranes assigned at this stage as shown in Fig. 3.7. (number of cranes C = 0 for RoRo ships at berths 2,3,4,and 5).

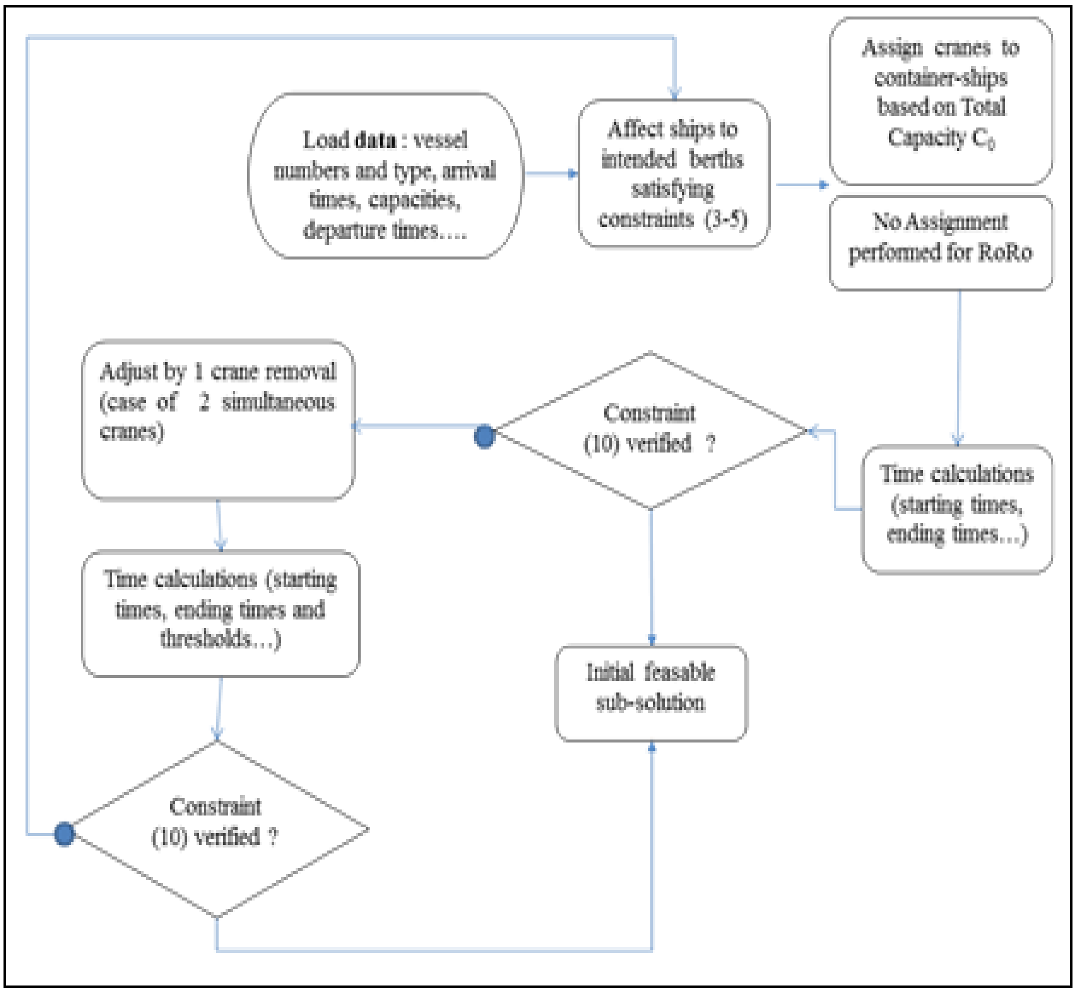

Figure 3.7 – Framework for the first sub-solution heuristic

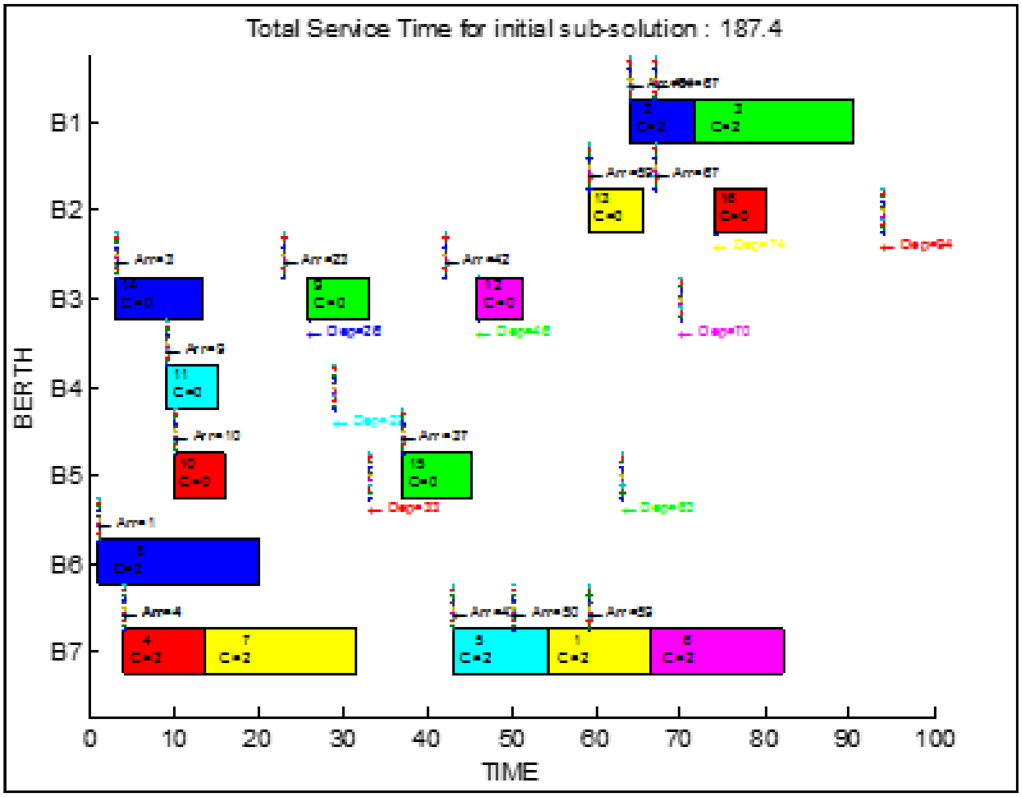

Figure 3.8 – Gantt chart for sub-solution

3.5.2 Steps 2: Event-Based construction heuristic

After finding the first sub-solution, an event based heuristic transforms it into a feasible solution with variable - in-time assignment. To better introduce the heuristic, such concepts, as State and Event have to be explained. A state is the status of an object, (a ship in our case), while an event is a thing that happens or takes place at a specific point in time. Thus, the state of a ship can be changed by an event such the Ending that change the ship from being served by h cranes to a ship that does not need any crane.

The execution order of the proposed event based heuristic is presented in Figure 8. Once the sub-solution found, the heuristic scans the events (found after times calculation), and decides if a re-assignment is necessary.

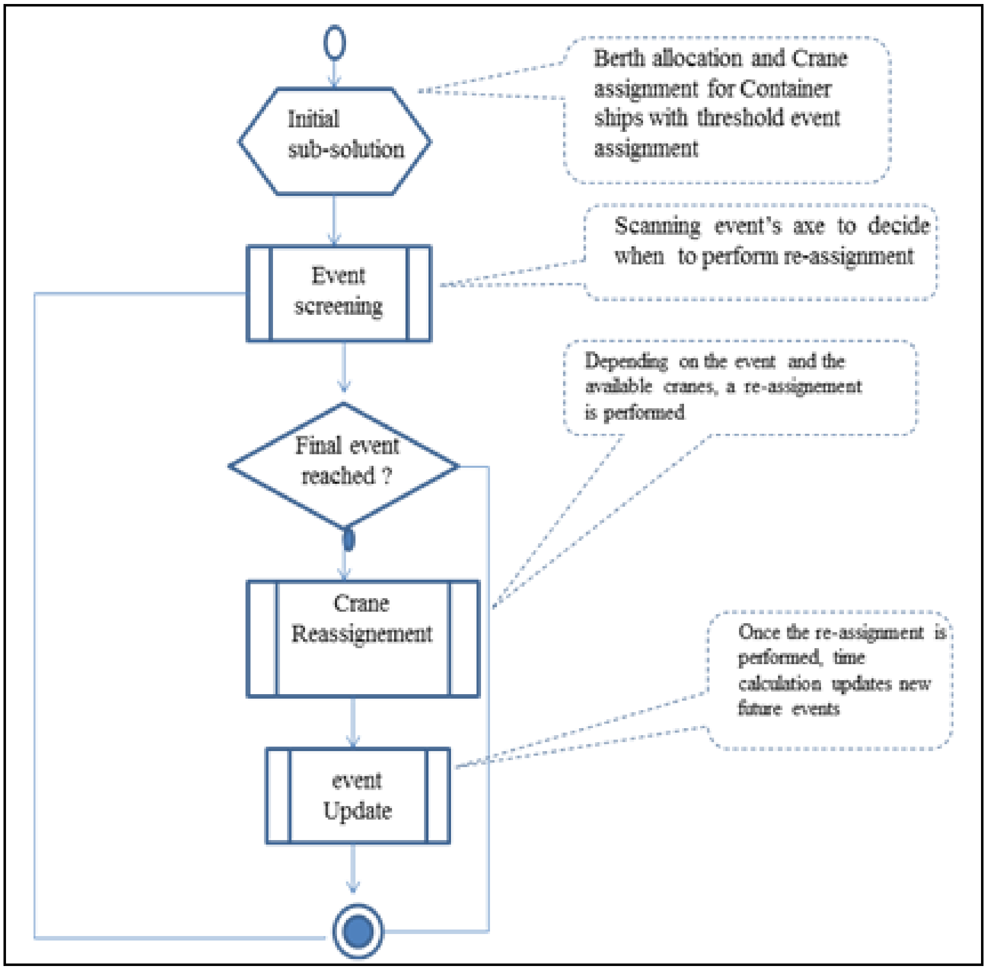

Figure 3.9 – Construction heuristic Crane re-assignment based event

A complete solution expected for our problem is a complete berth allocation and crane assignment for all incoming ships. Container ships can be served by, at most, 2 cranes simultaneously and just one crane is needed for handling the containers aboard the RoRo-ships. Consequently, RoRo ships can start being served, once berthed, by trucks to load/unload the trailers aboard. Container-ships are being « served » only when their cranes are assigned.

By scanning events, the heuristic decides whether it is necessary or not to execute a re-assignment to ships. The initial sub solution has already a partial crane assignment but the heuristic tries to modify and complete it. Figure 3.9 illustrates the main events in our problem.

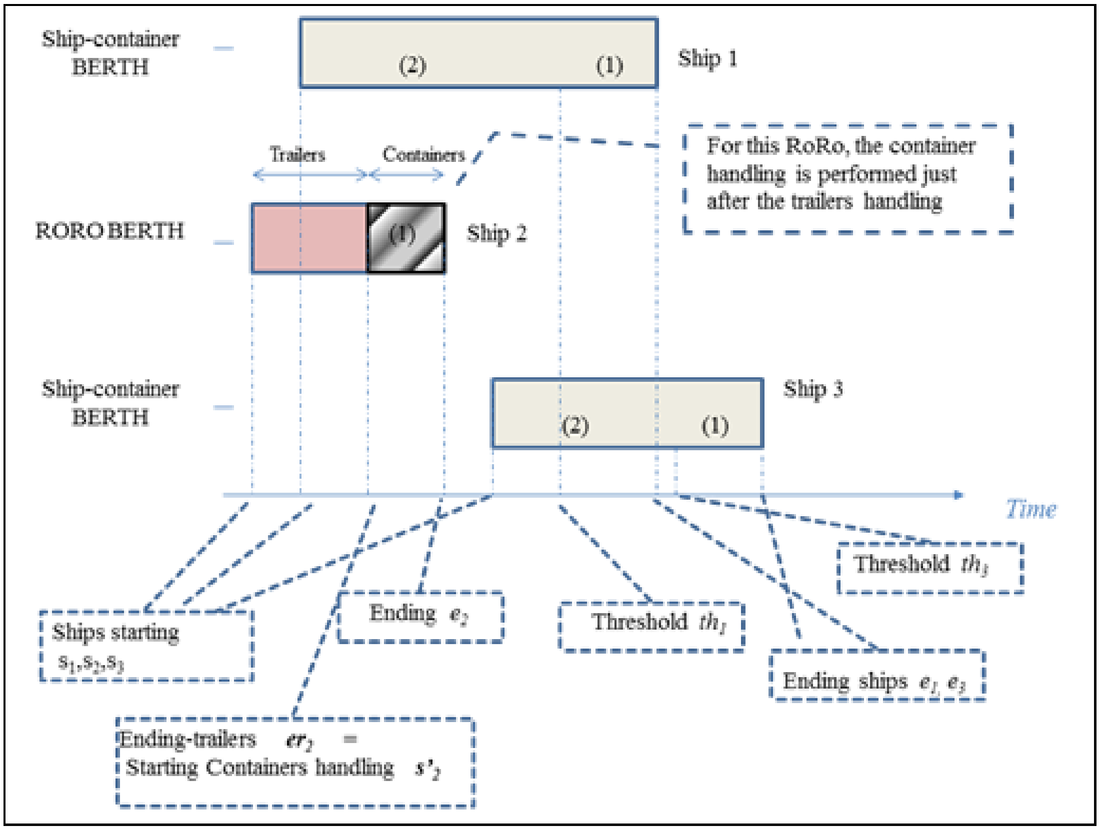

Figure 3.10 – Main Events

Events relating to container ships are starting (s_i), the threshold (th_i), and ending (e_i). Events relating to RoRo ships are starting (s_i), starting horizontal charging service(s'_i), ending horizontal charging service (er_i), danger zone (dz_i), and ending (e_i).

When any event occurs, the algorithm has to check the available cranes at that moment and decides how to re-assign cranes if necessary. For example, for the containerships, a threshold is an event for which the number of cranes decreases from 2 to 1 because the number of remaining containers does not need 2 cranes simultaneously. It is adequate to assign the crane left to another ship. Another example, when a container ship is being served by 2 cranes and another container ship is berthed (and no available cranes), a transfer is possible from ship 1 to ship 2 to avoid waiting.

For the RoRo assignment and since the containers handling is not necessary at the beginning of RoRo service, we add an event called danger zone (dz), it is the late starting time to handle the container aboard since the RoRo ships have a due time, and leaving after this due time causes a delay which must be avoided as much as possible. Consequently, if no crane is assigned to RoRo ship until its danger zone, a transfer from another container ship is necessary. If cranes are available before this event, the RoRo could accept a crane transfer.

In Figure. 3.10, some cases of re-assignments are presented, the Gantt chart is the transformed assignment of the incoming ships illustrated previously by their sub-solution in Figure. 3.7. The threshold event for containerships changes the assignment at one crane, and all the RoRo-ships are then served by 1 crane when it is available. RoRo ships number 14 (berth 3) and 10 (berth 5) have received their crane assignment at respectively the threshold events for containerships 8 and 4.

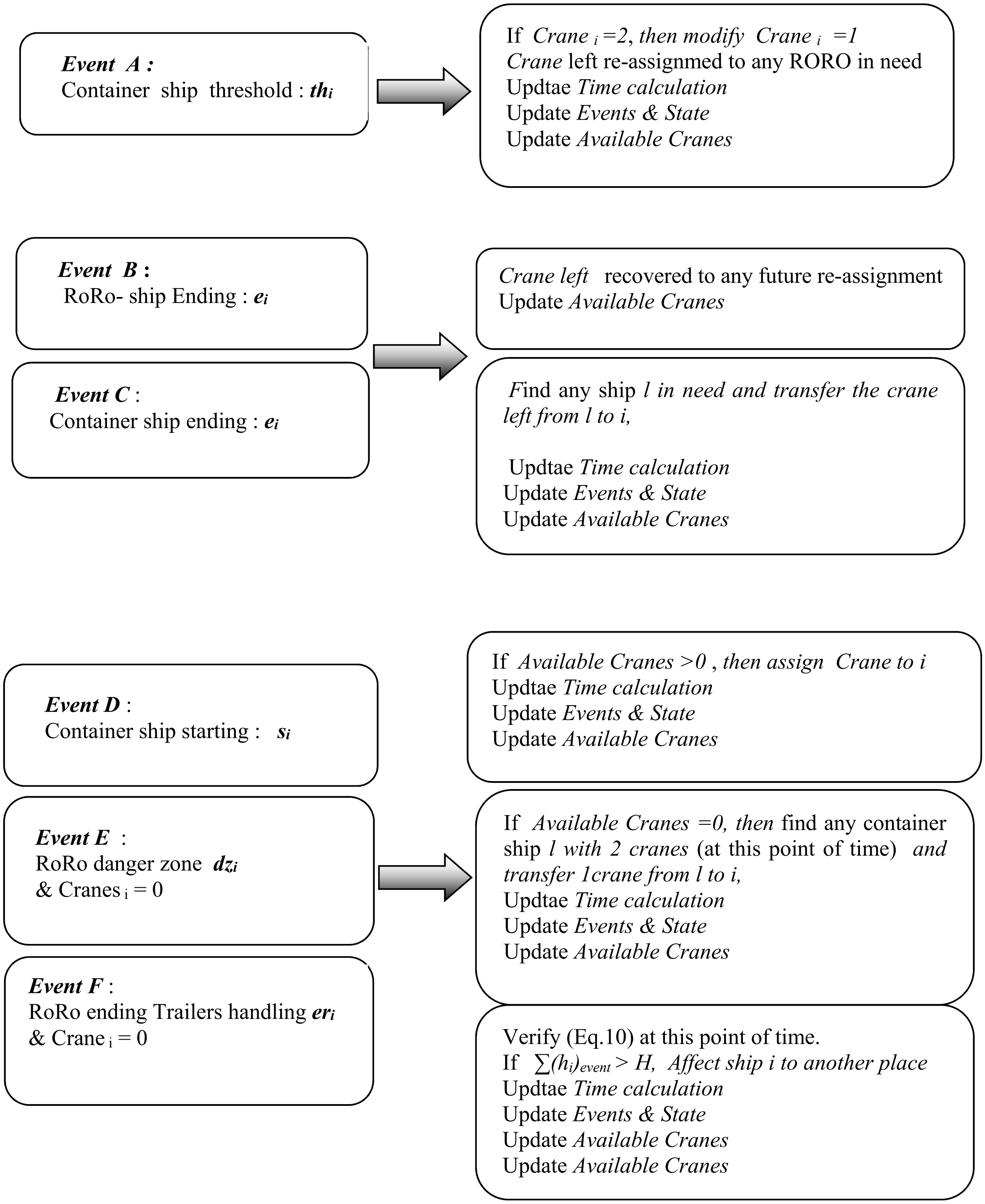

Figure 3.12 – Cranes Re-assignement scenarios

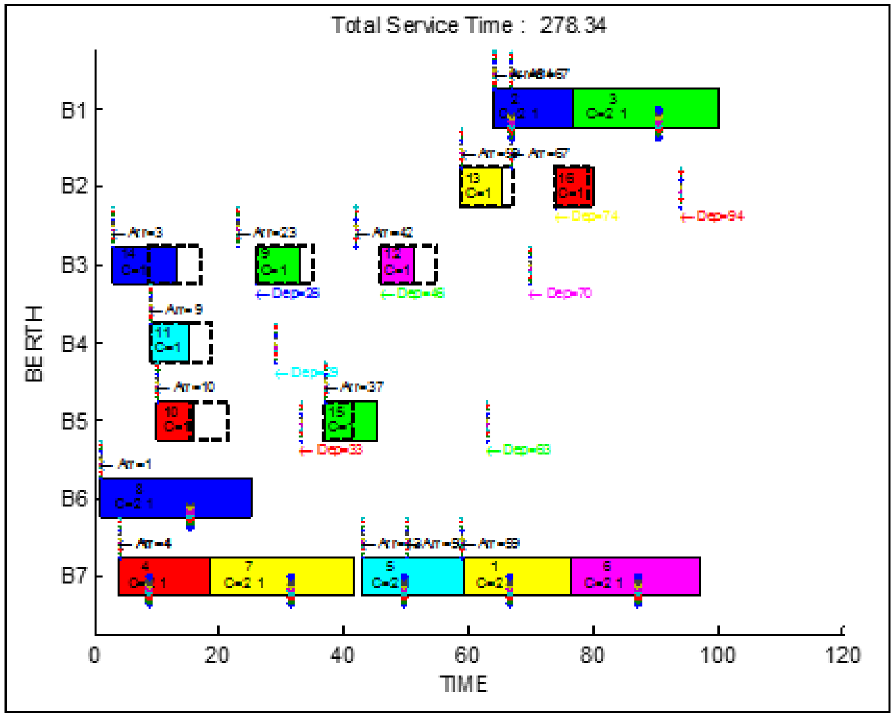

Figure 3.13 – A complete feasible solution after re-assignment.

The most interesting decision the algorithm can consider is the crane transfer between Containership and the RoRo ship when the latter needs to handle its containers onboard.

Figure 13 is presenting this case. Containership 2 berthed in position B6, has transferred one of its two cranes to RoRo ship 11 at berth 2, because at this moment, RoRo ship is entring to its danger-zone, so it was very important to transfer to it a crane from a container ship using 2 cranes since no other cranes are available. In the Gantt chart, and for the visualization of the transfer, the number of cranes assigned to container ship 2 during the service is represented by [C=2 1 1], that means that the transfer has happened before the threshold th_2

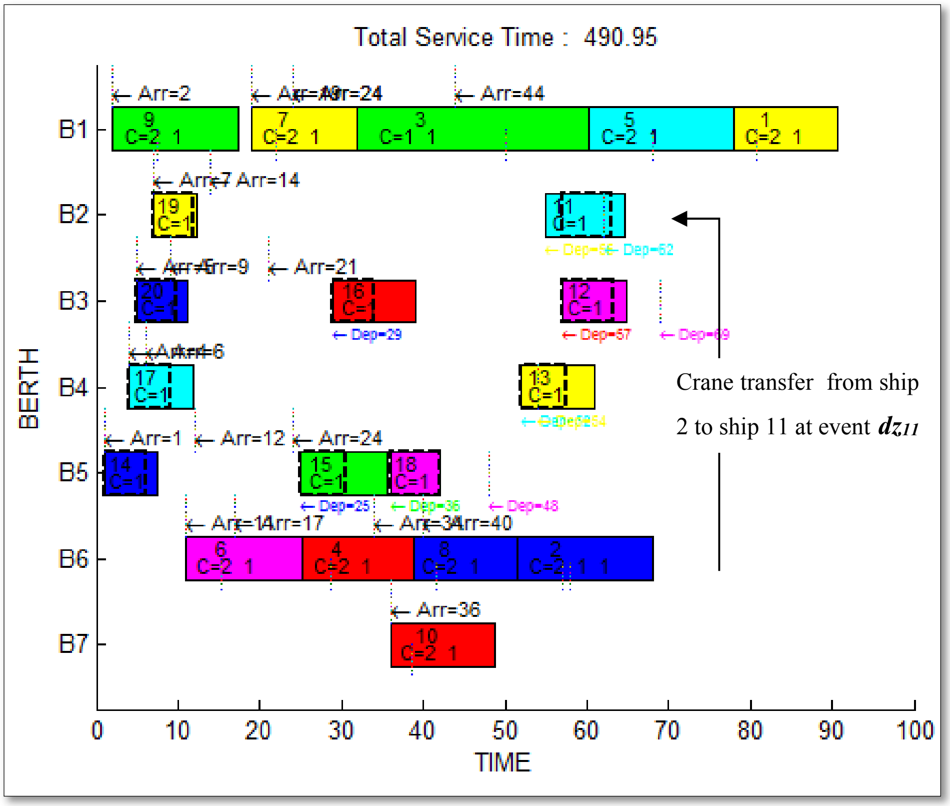

Figure 3.14 – Case of Gantt solution considering crane transfer

3.5.3 Step 3: METAHEURISTIC for near Optimal Solution

After finding an initial feasible solution using the event-based construction heuristic, a near optimal solution process search starts. In this study, two meta-heuristics are used to compare the efficiency and the performance of a population meta-heuristic versus a local point search meta-heuristic algorithms in solving this kind of issue.

Artificial Bee Algorithm

An ABC, which is an optimization algorithm based on the intelligent behavior of honey bee swarm proposed by Karaboga and Basturk in 2007. The ABC has been one of the most often applied in the past few years Swarm Intelligent algorithm to solve many different types of applications. It is a population-based algorithm for combinatorial optimization that is inspired by the foraging behavior of bees. It mimics the colony behavior to search for the best source of food. For further details, the reader is referred to (Karaboga and Basturk, 2007)

The general algorithmic structure of the ABC optimization approach is given as follows:

Initialization Phase
REPEAT
Employed Bees Phase
Onlooker Bees Phase
Scout Bees Phase
Memorize the best solution achieved so far
UNTIL (Cycle = Maximum Cycle Number)

In ABC, the position of a food source represents a possible solution (initial feasible solution) to the problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. In the basic form, the number of employed bees is equal to the number of food sources (solutions) since each employed bee is associated with one and only one food source. The main detailed steps of the ABC algorithm implemented for this BACAP resolution are presented as follow:

Algorithm 1
15. Initialize parameters. 16. Generate the initial population of solutions x_i randomly which contain *NS* solutions (number of sources), in our case, a solution is a feasible berth and crane assignment plan. {**initialization**} . 17. Evaluate the fitness function $f(x_i)$ of all solutions in the population, where $f(x_i)$ is $\frac{1}{1+Total\ Service(x_i)}$. 18. Keep the best solution x_{best} in the population. {**Memorize best solution**}. 19. Set cycle =1 **20. Repeat** 21. Generate a new neighborhood solution υ_i from the old solution x_i performing a little perturbation to the berth and crane plan. {**Employed bees**}. 22. Evaluate the fitness function $f(v_i)$ for all solutions in the population. *23.* Keep the best solution between current and candidate solutions {**Greedy selection**}. *24. C*alculate the probability P_i, for the solutions x_i, where $P_i = \frac{f_i}{\sum_{i=1}^{NS} f_i}$. *25.* Generate the new solutions v_i (neighborhood) from the selected solutions depending on its *Pi* *{***Onlooker bees***}*. 26. Evaluate the fitness function *fi* for all solutions in the population. *27.* Keep the best solution between current and candidate solutions *{***Greedy selection***}*. 28. Determine the abandoned solution if exist, replace it with a new randomly solution x_i {**Scout bee**}. 14: keep the best solution x_{best} found so far in the poupulation . 15: *cycle=cycle+1* **Until** cycle ≤ cycle max

As stated earlier, the ABC is an iterative procedure to search for improvements in initial feasible solutions to achieve a near-optimal solution towards the end of its research cycles. The starting point will, therefore be a population of feasible solutions (complete feasible solution after the event-based heuristic). Thereafter, in the Employed Bees phase, a neighbor is generated for each solution of the initial population via the neighborhood procedure.

The neighbor that is better than its associated solution is retained, otherwise, the old solution is kept. Before proceeding to the next generation, the Scout Bees phase is performed to replace solutions that have not been improved during the process. The iterations follow one another until the maximum number of cycles is reached.

The proposed framework of ABC applied to our dedicated berth allocation and crane variable-in-time assignment is presented in Fig. 3.14.

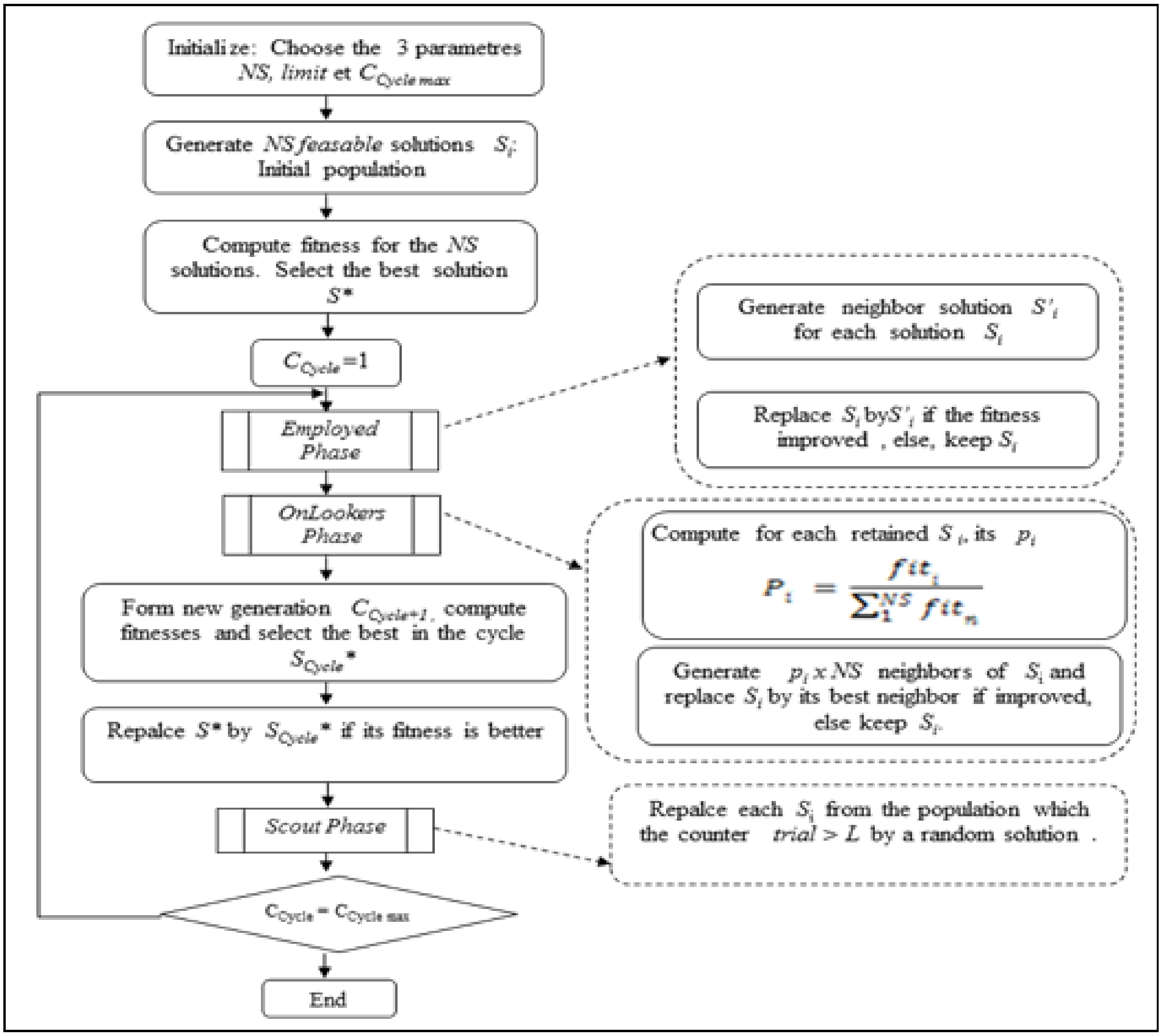

Figure 3.15 – Artificial Bee Colony Algorithm adopted for the problem

In our knowledge, the Artificial Bee Colony has not been proposed yet to solve a berth and crane assignment problem, it will be interesting to test the performance of other population based algorithm to such issue apart from, the most often one in the past years used one and its variants, the Genetic Algorithm.

Extended Great Deluge

The second algorithm chosen is a local search or neighborhood meta-heuristics. To that end, we applied the Extended Great Deluge (EGD) algorithm proposed by Burke et al. (2004). The EGD algorithm based on a neighborhood search accepts every solution whose objective function is less than or equal to an upper limit (level) B or less than a current solution. The value of B is monotonically decreased during the search and bounds the feasible region of the search space. The advantage of this method is that only one input parameter, called ΔB (cf the following algorithm in Table 2), which is the decay rate at each step, has to be adjusted. According to (Burke et al., 2004), founder of the method, this parameter can be interpreted as a function of expected search time and expected solution quality, which are relatively easy to specify.

.

Algorithme 3.1 – Extended Great Deluge algorithm

Set the initial solution ***S***
Calculate initial cost function ***f(s)***
Initial ceiling ***B=f(s)***
Specify input parameter Δ***B=?***
While not stopping condition do
Define neighbourhood ***N(s)***
Randomly select the candidate solution S^* Є ***N(s)***
If (f(s*) ≤ f(s)) or (f(s*) ≤ B)
Then Accept S*
Lower the ceiling B = B - ΔB
End while.

Improvement regarding initial solution is carried out through implementation of the EGD algorithm presented in Table 2. As mentioned above, it uses a boundary B, which is initially set equal to the initial solution, and is reduced gradually through the improvement process.
For the application of this algorithm to our problem, we needed:

- The initial solution S found by the constructed heuristic.
- The definition of the neighborhood N(S) of this solution.

The neighborhood was created while making minor modifications to the initial solution S, such as to the permutation between two container ships taken randomly.

The event based heuristic is then performed to re-assign cranes since the perturbation has probably affected the solution feasibility associated to constraint (3.9). Following the modifications, the algorithm applied tests on the neighborhood solution to check if all the constraints have been fulfilled.

This metaheuristic was used in (El Asli and al. (2016)) to solve a dynamic berth and crane scheduling problem proposed by Liang and al. (2009) and provides an improvement in results compared to the genetic algorithm variants resolution for the same data set used by Liang an al. (2009) and Ma and al. (2015).

Simulation neighborhood Example

The neighborhood concept applied is the same for both ABC and EGD. An initial feasible solution with the complete variable in-time crane assignment is mutated by making a permutation between two containerships. Thereafter, the event based heuristic is applied to that neighbor to transform it in an other feasable solution.

In the next Figures, 3.15 and 3.16, an example of this transformation is illustrated by respectively 15 ships instance's solution and its neighbor's Gantt charts.

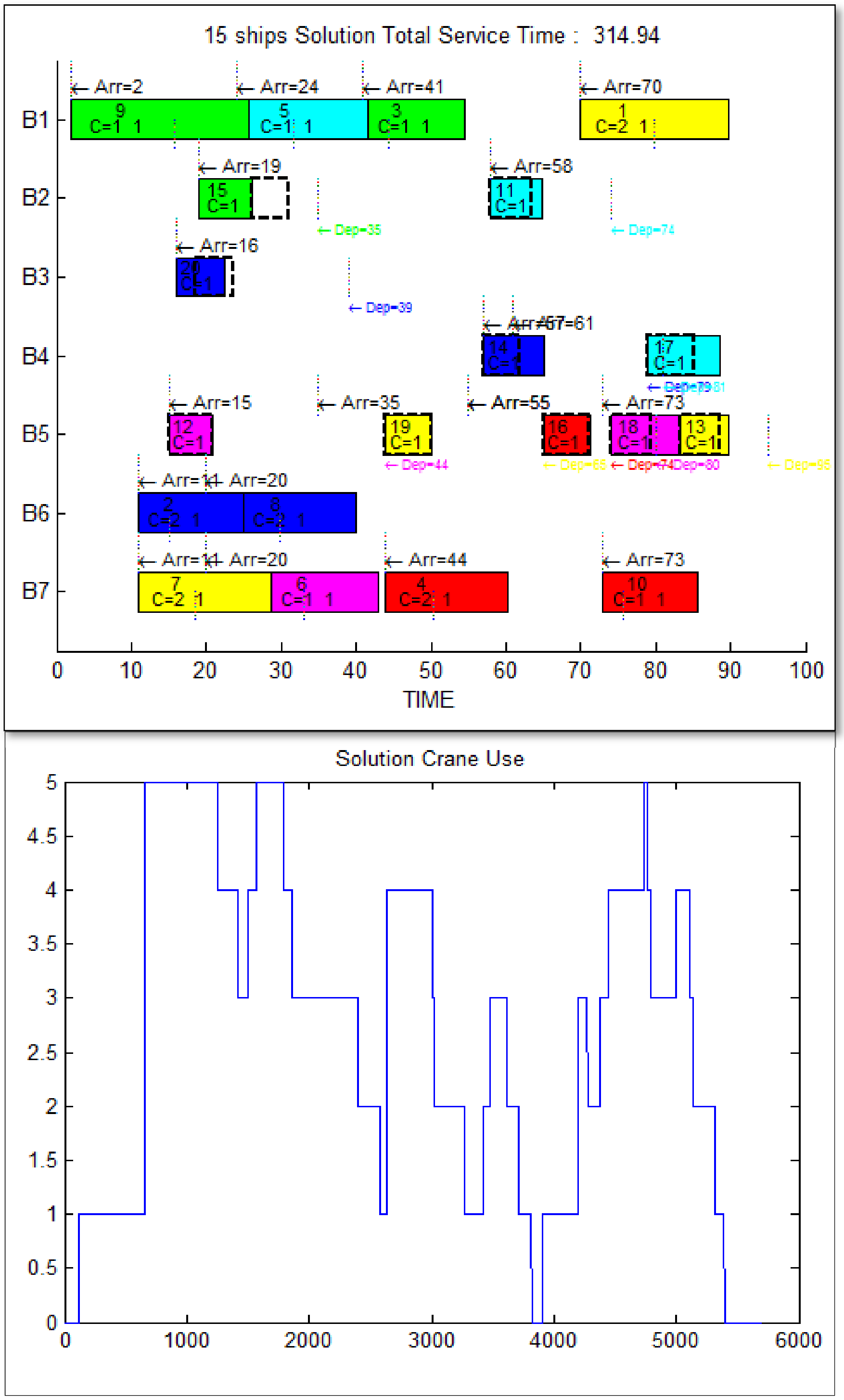

Figure 3.16 – 15 ships Solution Gantt Chart and
its crane use during time horizon

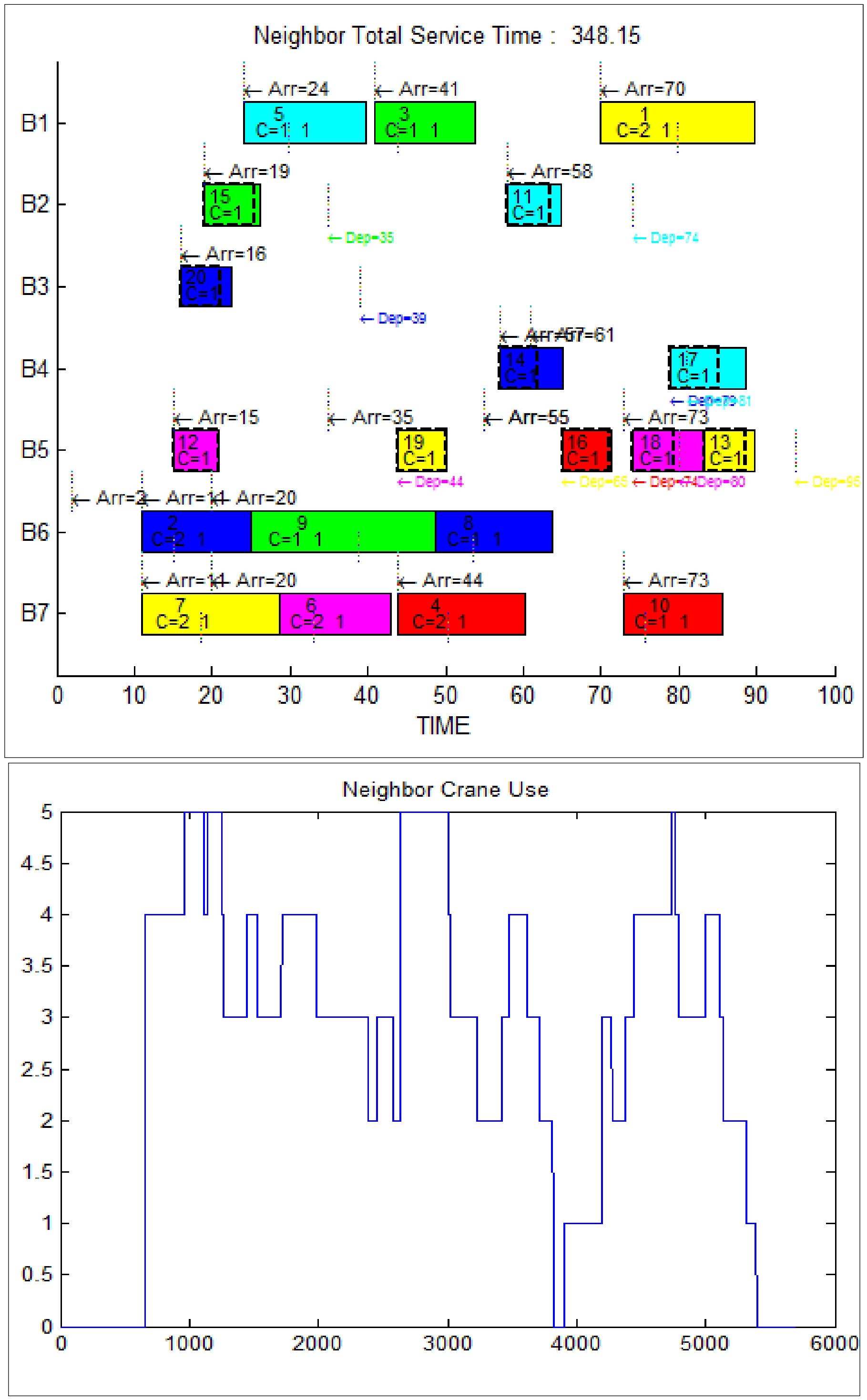

Figure 3.17 – Neighbor Solution Gantt chart
and its crane use during time horizon

The illustration of neighborhood shows for this example the mutation of the containership from the berth 1 to the berth 6. Crane assignment also changed for the RoRo 15 and 20 consequently to respect the constraint (10) constantly. The graphs at the right of Gantt charts present the use of available crane along the time horizon, it is clear that in any time, the number of maximum cranes used simultaneously does not exceed 5. For that example, the neighbor will not be retrained, in fact, its total service time (348 h) is greater than the initial solution's Total service time (314), the algorithm will search another neighbor then.

3.6 Computation experiments and discussion

The dedicated berth and crane variable in time assignment has been implemented in Matlab language and the experiments have been carried on a PC with an Intel Pentium 2.2 GHz CPU and 4G DRAM.

This section presents some computational experiences considering problems of different sizes that range from 20 up to 35 vessels in order to reach performance conclusion regarding the instances.

The experiments concern a terminal with 7 berths, 3 of which (number 1, 6 and 7) are for container ships and 4 (number 2, 3, 4 and 5) are intended for RoRo. The number of cranes available at the terminal is 5.

Crane speed is set to 20 containers/hour and trailer charging/discharging speed is set to 30 trailers/hour. The two interference coefficients $coef_1$ and $coef_2$ are set respectively to 1.5 and 1.2.

3.6.1 DATA generation for experiments

To simulate the simultaneous berth and crane assignment process, ship data for experiments are artificially generated. Arrivals for the two types of ships are generated by the discrete uniform distribution U[0,120]. Time windows for the RoRo ships are varying between 24 and 48 hours. Departure time for RoRO are then generated by Arrival Times + U [24,48]. Containers Capacities for containership and RoRo are generated respectively by U[300,800] and U[70,100], and finally trailers capacity for RoRo is obtained by U[150,400].

The planning Horizon is set to one week (168 hours). It is important that for each instance generated, the maximum container handling capacity is not exceeded, it is then assumed that the terminal will not handle more than its maximum handling capacity. This maximum capacity is determined by Eq.(20) as follows:

$$\begin{aligned}&Max\ Handling\ Capacity\\&= Total\ Cranes\ productivity\ x\ duration\ of\ the\ planning\ horizon.\end{aligned} \tag{3.20}$$

For the case of this work, crane productivity is 20 Containers/hour, consequently, the maximum handling capacity will be , 5 cranes x 20 cont. / hour x 168 hours = 16 800 container.

3.6.2 Experimental results

Five experiments have been executed for each size problem. Each experiment is run 5 times to record the best total service time. The same data set for each simulation is used for comparison of the ABC and EGD results. As presented previously, size problem ranging from 20 to 35 with an increment of 5 are simulated. Increasing the number of ships, leads to exceeding maximum capacity handling defined by Eq.(3.20)

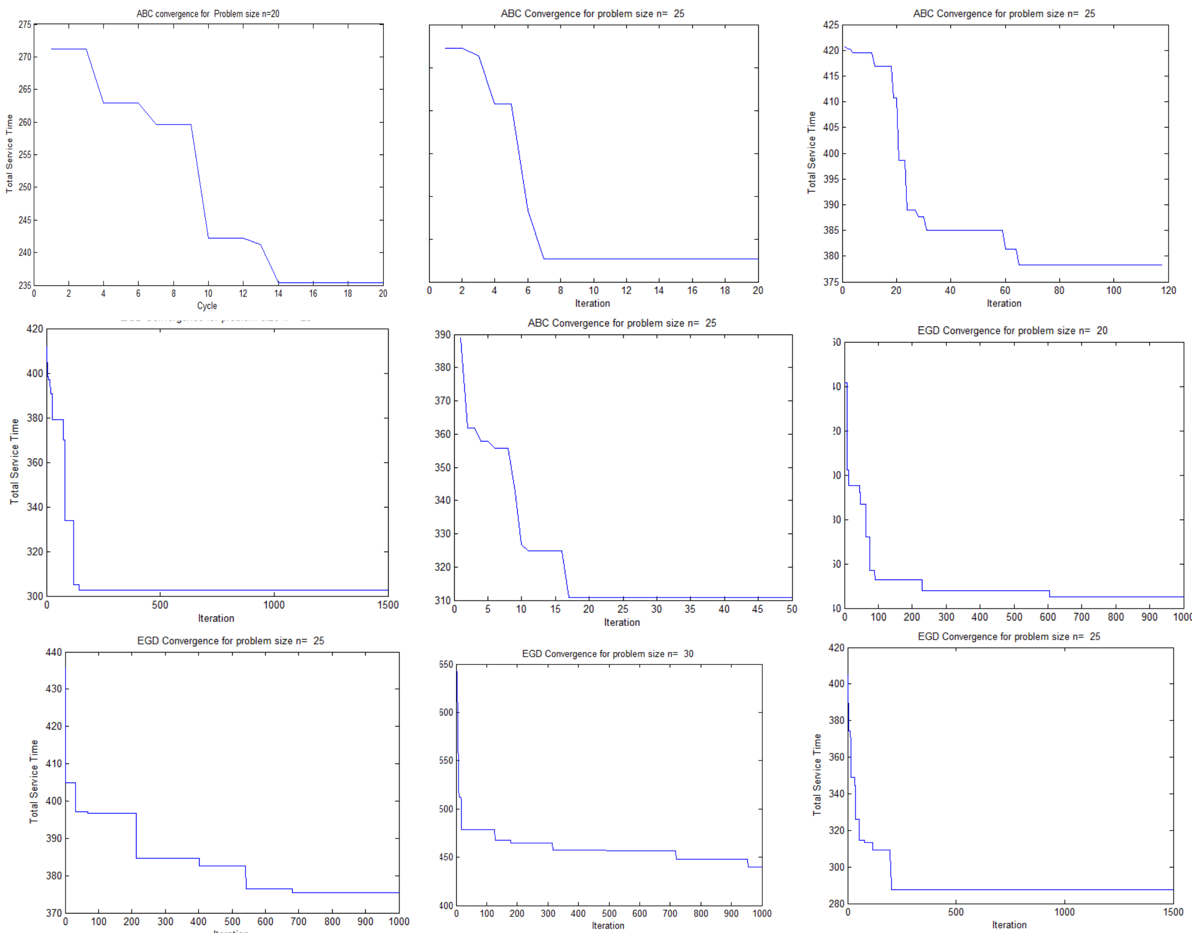

Figure 3.18 – Convergence curves behaviour for both algorithms

The parameters chosen for tuning are those providing the best results in our test runs in reasonable computational times. Especially for ABC Cycle max and EGD Iterations, several trials have been performed to decide on the values that will be chosen. Fig. 17 shows some convergence curves for both meta-heuristics. The decision was taken by making a compromise between calculation time, curve behaviour through iterations and the variance of the results of the same data instance during a set of executions, which is, for us, a convergence indicator.

The aim of this section is to compare the results collected by running EGD and ABC on different problem sizes.

Table 3.2 records the obtained objective function value Z, representing the total service time, and Time computation for each meta-heuristic. In the last column, is given the relative error calculation of ABC against EGD. We notice the long time computation for both algorithms, with regard to few seconds meta-heuristic's usual calculation times. It is thus a new way of improving for future work.It can be seen that EGD and ABC results are comparable for these conditions tuning. The close computation time might suggest that the number of visited solutions is the same for both algorithms.

Another interesting finding appears in observing the gap between Z values and even times computation between 30 and 35 ships problem size. In fact, the difference seen for Z value is not only for increasing handling times but also for increasing waiting and thus delay times.

Tableau 3.2 – ABC and EGD Results for test instances

Problem size		**ABC :** NS=50,limit=30, Max_{Cycle}=50		**EGD** :1500 iterations, *deltaB*=0.05		**Comparison**
		(1) Z (hour)	Computational Time (sec.)	(2) Z (hour)	Computational Time (sec.)	$\frac{(1)-(2)}{(2)}\ x\ 100\%$
20 ships : 10 containerships 10 RoRo.	5 runs x					
	1	235.43	161	228.8	134	2.8
	2	266.56	119	268.32	150	-0.6
	3	256.61	167	244.95	135	4.7
	4	298.53	158	304.5	147	-1.9
	5	243.72	149	240.22	154	1.4
25 ships: 10 containerships 15 RoRo.	5 runs x					
	1	337.63	260	328.42	245	2.8
	2	413.73	230	408.20	214	1.35
	3	378.46	240	375.47	256	0.79
	4	310.90	215	306.4	207	1.4
	5	396.78	222	402.87	234	1.5
30 ships: 15 containerships 15 RoRo.	5 runs x					
	1	439.62	345	450.40	330	-2.39
	2	456.34	365	447.89	356	1.88
	3	423.04	322	447.34	310	-5.43
	4	440.89	369	445.23	379	-0.9
	5	470.67	312	459.11	306	2.5
35 ships: 15 containerships 20 RoRo.	5 runs x					
	1	715.74	1068	710.45	998	-2.39
	2	730.34	1156	738.23	1053	1.88
	3	708.67	1043	700.54	1022	-5.43
	4	756.90	1240	770.23	1324	-0.97
	5	722.87	1096	710.33	985	2.51

3.7 Conclusion

In this study, a new nonlinear MIP Model for the simultaneous discrete berth allocation and crane assignment problem with the variable-in-time-crane assignment strategy for a special intended berths container terminal has been formulated. A rich event-based construction heuristic to find feasible solutions has been proposed. Improved solutions are then provided through an optimum search process based on meta-heuristics. The Extended Great Deluge, a local search algorithm has been compared to Artificial Bee Colony, which is based on the intelligent foraging behaviour of honey bees swarm. Computational comparisons, for the problem sizes taken and tuning parameters show that both algorithms do not significantly differ in solutions. Other topics for future research may be considered such as incorporating other realistic factors such as crane transfer time between ships, Terminal breaks and pre-emptive tasks in handling container vessels. Computational times are subject to improving by some programming code optimizations.

Acknowledgement

The author would like to acknowledge the generous assistance and valuable information provided by the authority and the employees of the Anonymous Container Terminal.

Conflict of interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

CONCLUSION

Ce travail a débuté par découvrir le monde du transport maritime et les problèmes logistiques auxquels sont confrontés les ports en général et les terminaux à conteneurs en particulier. La problématique du *BERTH ALLOCATION* était la première à être découverte. Le but, au début de cette recherche, était de lui appliquer les méthodes approchées d'optimisation pour sa résolution puisque le problème se classait NP-complet. Avec nos lectures et notre approfondissement dans cette première problématique, il s'avérait indispensable de l'intégrer au problème de *CRANE ASSIGNMENT* puisqu'ils étaient intimement liés comme expliqué dans notre état de l'art. Le modèle discret dynamique de référence a été étudié en première étape et résolu avec une méta-heuristique de recherche locale : LE GRAND DELUGE ÉTENDU. La comparaison des résultats avec la variante de l'algorithme génétique appliqué au problème originel a montré une amélioration de 4 % sur une instance de 11 navires. Ce qui nous a conduit à mieux explorer les performances de cette méta-heuristique en l'appliquant à des instances plus larges et en comparant les résultats au recuit simulé, qui est la méta-heuristique de recherche locale qui lui est la plus semblable. Le chapitre-article 1 traite cet aspect. La contribution la plus enrichissante durant cette première partie de la thèse est la proposition de deux (2) variantes multi-objectif de l'algorithme du grand déluge étendu, puisque nous n'avions pas trouvé dans la littérature un *Extended Great Deluge* pour l'optimisation multi-objectif. Les deux étant basées sur le principe de non-dominance de *PARETO*. L'une pondère les objectifs pour n'en faire qu'un seul, *PA-WEGD* et l'autre, les traite simultanément à chaque itération, *PA-EGD.*

La curiosité de découvrir la réalité du *BERTH ALLOCATION AND CRANE ASSIGNMENT* nous a poussé à appliquer pour un stage au terminal Tunisien de Rades pour voir de plus près les contraintes du problème. Le stage au terminal tunisien était une occasion très fructueuse pour proposer notre deuxième originale contribution dans ce travail de recherche : formuler le modèle non linéaire de la problématique complexe du BACAP à RADES. Les deux chapitres-articles 2 et 3 présentent les deux modèles avec chacun une stratégie différentes d'assignation de grues ; *invariable-in-time* et *variable-in-time assignment.*

Et comme méthode de résolution, nous avons opté pour les algorithmes de colonie d'abeilles artificielles qui n'ont pas été appliquées pour ce genre de problèmes dans la littérature.

Nous ne pouvons pas, à la fin, nous empêcher d'ouvrir des perspectives très prometteuses à ce travail, à savoir :

- Optimiser l'algorithme de résolution de l'assignation variable des grues pour minimiser le temps de calcul.
- Construire une base de Benchmarks de la problématique de Radès pour comparer les résultats des diverses méthodes de résolution.
- Valider les performances des deux variantes EGD-multi-objectif sur d'autres problèmes d'optimisation.
- Tenir compte des autres complexités de la réalité du terminal de Radès pour bâtir d'autres modèles plus réalistes (par exemple, considérer l'aspect stochastiques des pannes des grues, prédire à l'avance le taux de chargement/déchargement des grues en considérant les différents attributs qui l'influencent…).

ANNEXE I

Le Terminal Tunisien de Radès

Le port de Radès occupe une place importante dans la chaîne de transport national de par sa spécialisation dans le trafic de conteneurs et unités roulantes (essentiellement le trafic des remorques). Crée en 1987, le Terminal de Radés assure plus que 95 % du trafic national en conteneurs et environ 96 % du trafic roulier.

Tableau A.1 – Caractéristiques des sections de quai du terminal de Radès

Poste à quai	Longueur en mètre	Tirant d'eau (mètre)	Type navires	Marchandises
1	150	9	Porte-Conteneurs	Conteneurs
2	150	9	Rouliers	Conteneurs et Unités roulantes
3	150	9.15	Rouliers	Conteneurs et Unités roulantes
4	150	9	Rouliers	Conteneurs et Unités roulantes
5	150	9	Rouliers	Conteneurs et Unités roulantes
6	200	9	Porte-Conteneurs	Conteneurs
7	180	9.15	Porte-Conteneurs	Conteneurs

La STAM (Société Tunisienne d'acconnage et de manutention), premier opérateur de conteneur en Tunisie, opère depuis Janvier 2005 au port de Radès en qualité d'Entrepreneur de manutention concessionnaire unique du terminal.

Avec sept (7) postes à quai d'une longueur totale avoisinant 1130 mètres, une superficie de 50 hectares, le terminal de Radès a traité 1270 navires en 2013 pour assurer 43 % du trafic global des marchandises traitées par son unique concessionnaire pour la manutention. Ce qui représente également 70 % de la marchandise conteneurisée du pays.

Tableau A.2 – Engins de Manutention pour la manipulation conteneurs et unités roulantes au terminal de Radès

Type d'engins	Nombre	Types d'opérations réalisées
Grues portuaires mobiles sur pneus	5	Chargement/déchargement des conteneurs
Cavaliers Gerbeurs	21	- Stockage des conteneurs dans les aires de stockage - Livraison des conteneurs
Reach Stackers	9	Transfert des conteneurs débarqués vers les RoRo-trucks Chassis.
Elévateurs pour conteneurs vides	15	Stockage
Chariots Élévateurs 15 à 45 t	7	Stockage
Chariots Élévateurs 3 à 10 t	6	Stockage
RoRo truck-Châssis	49	Transfert des conteneurs débarqués vers les entrées des aires de stockages où le cavalier gerbeur se chargera du stockage

Spécificité des opérations du terminal de Radès

Comme présenté ci-haut, le terminal à conteneurs de Radès accueille aussi des navires spécialisés appelés Rouliers ou RoRo (Roll on Roll off), qui sont chargés de remorques (conteneurs sur châssis avec roues pour faciliter le transfert) ainsi que de conteneurs classiques sur leur pontée. Ces RoRo leur sont affectés des postes dédiés (zones de quai N° 2, 3, 4 et 5). Le chargement/déchargement des remorques se fait par des camions-remorques (RoRo-trucks), et les conteneurs sont manutentionnés par les mêmes grues utilisées pour les porte-conteneurs. Ce partage de ressources (grues) est la cause d'une chute de rendement dans la manutention des conteneurs.

Mise à part cette première contrainte, les RoRo sont contraints par une fenêtre de temps (temps d'arrivée et temps de départ) alors que les porte-conteneurs arrivant au port de Radès n'ont pas une date de départ imposée. En effet, ce sont des feeders (un feeder est un navire de petit tonnage qui effectue le pré et le post transport de conteneurs vers des ports où n'escale pas le navire mère de lignes régulières), qui arrivent vers le port de Radès avec, exclusivement, des conteneurs pour la Tunisie, donc ne sont pas obligés de repartir selon une échéance pour une escale ultérieure dans un autre port.

L'objectif des gestionnaires serait donc de minimiser le temps total d'attente en rade (La rade est un plan d'eau marin permettant le mouillage d'une flotte) des porte-conteneurs et le retard de départ des Rouliers. Ces deux mesures de performances sont intimement liées au service de manutention qui lui aussi devrait être géré d'une façon optimale. Pour la situation actuelle, la manutention au port se fait systématiquement en suivant la règle du premier arrivé, premier servi, en essayant de respecter la date de départ des navires Rouliers et en partageant les grues au besoin. Il n'a y a pas de procédures exactes pour ce partage, c-à-d, que le déplacement des grues entre porte-conteneurs et RoRo se fait à n'importe quel moment durant un quart de travail (shift de travail) en interrompant, si nécessaire, la manutention du porte conteneur puisque le Roulier est contraint par la date de départ imposée. Cette interruption causerait nécessairement un retard pour le service du porte-conteneurs et donc une attente prolongée des navires en rade.

De ce fait, une meilleure politique pour la planification et l'assignation des grues est demandée. Cette problématique a été abordée dans les chapitres-articles 2 et 3 de cette thèse.

LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES

Agra, A., Oliveira, M. (2016). MIP approaches for the integrated berth allocation and quay crane assignment and scheduling problem. European Journal of Operational Research, May 01, 2016

Arango, C., Cortès P., Munuzuri J., Onieva L., (2011). Berth Allocation planning in Seville inland port by simulation and optimisaton. Advanced Engineering Informatics, 25, pp.452-461

Asim A. R. El Sheikh, Ray J. Paul, Alan S. Harding, David W. Balmer, (1987). A Microcomputer-Based Simulation Study of a Port. The Journal of the Operational Research Society, Vol. 38, No. 8, Current Simulation Research, pp. 673-681

Bierwirth, C., Meisel, F., (2010). A survey of berth allocation and quay crane scheduling problems in container terminals. European Journal of Operational Research. No. 202, pp. 615-627

Bierwirth, C., Meisel, F., (2015). A follow survey of berth allocation and crane scheduling problems in container terminals. European Journal of Operational Research. 1-15.

Burke E.K., Y. Bykov, J.P. Newall and S. Petrovic, (2004). A Time-predefined local search approach to exam timetabeling problems. IIE Transactions, vol. 36 (6), 2004, pp. 509-528.

Chang D, Jiang Z, Yan W, He J., (2010) Integrating berth allocation and quay crane assignments. Transportation Research Part E 46 (2010) 975–990

Cheong, C.W, Tan,K.C. (2008). A multi-Objective Multi-Colony Ant Algorithm for Solving the Berth Allocation Problem. Studies in Computational Intelligence. 116, 333-350

Chunxia Yang , Xiaojun Wang, Zhenfeng Li.2012. An optimization approach for coupling problem of berth allocation and quay crane assignment in container terminal. Computers & Industrial Engineering .Vol.63. 243–253

Cordeau, J-F., Laporte, G., Legato, P., Moccia, L., (2005). Models and tabu search heuristics for the berth-allocation problem. Transportation Science. 39, 526–538.

El Asli N, Dao T.H., & Bouchriha H., (2016), Extended Great Deluge Metaheuristic based Approach for the Integrated Dynamic Berth Allocation and Mobile Crane Assignment Problem. International Jouranl of Advanced Engineering Research and Application, Vol.2, Issue 5.

Garey,M.R, Johnson,D.S. (1979). Computers and Intractability: A guide to the theory of NP-Completeness. Freeman,San Fransisco,CA.

Giallombardo G., Moccia L., Salani M., Vacca I., (2010). Modeling and Solving the Tactical Berth Allocation Problem. Transportation Research Part B: Methodological .Volume 44, Issue 2, Pages 232-245

Guan, Y, Xiao, W-Q, Chueng, RK and Li, C-L. 2002: A multiprocessor task scheduling model for berth allocation: Heuristic and worst case analysis. Operations Research Letter 30: 343–350.

Guan, Y., Cheung, R.K., (2004). The berth allocation problem: Models and solution methods. Operational Research Spectrum 26, 75–92.

Hansen,P., Oguz,C. Mladenovic,N.(2006).variable neighborhood search for minimum cost berth allocation. European Journal of Operational Research.

Henesey,L., Paul Davidsson, Jan A. Persson. Using Simulation in Evaluating Berth Allocation at a Container Terminal

Han, XiaoLong , Gong, Xing; Jo, Jungbok (2105) , Computers and Industrial Engineering, v 89, p 15-22.

He, J,(2016) Berth allocation and quay crane assignment in a container terminal for the trade-off between time-saving and energy-saving. .Advanced Engineering Informatics, v 30, n 3, p 390-405.

Hendriks,M., Marco Laumanns ·Erjen Lefeber, Jan Tijmen Udding (2010), Robust cyclic berth planning of container vessels OR Spectrum (2010) 32:501–517

Hsu, Hsien-Pin(2016). A HPSO for solving dynamic and discrete berth allocation problem and dynamic quay crane assignment problem simultaneously. Swarm and Evolutionary Computation, v 27, p 156-168.

Hu,Z-H.,(2015) Heuristics for solving continuous berth allocation problem considering periodic balancing utilization of cranes. Computers & Industrial Engineering, v 85, p 216-26.

Hu Q.M., Hu Z H., Du Y., (2104). Berth and quay-crane allocation problem considering fuel consumption and emissions from vessels. Computers & Industrial Engineering 70 (2014) 1–10.

Huang J, Wang F, Shi N., (2014). Resource Allocation Problems in Port Operations: A Literature Review. Seventh International Joint Conference on Computational Sciences and Optimization.

Hassan, S.A. (1993),“Port activity simulation: an overview”, Simulation Digest, pp. 17- 36.

Imai, A., Nagaiwa, K., Chan,W.T.(1997). Efficient planning of berths allocation for container terminals in Asia. Journal of Advanced Transportation. (33) 75-94.

Imai, A., Nishimura, E., Papadimitriou, S., (2001). The dynamic berth allocation problem for a container port. Transportation Research B 35, 401–417.

Imai, A., Nishimura, E., Papadimitriou, S., (2003). Berth allocation with service priority. Transportation Research Part B 37, 437–457.

Imai, A., Sun, X., Nishimura, E., Papadimitriou, S., (2005). Berth allocation in a container port: Using a continuous location space approach. Transportation Research B 39, 199–221.

Imai, A, Nishimura, E, Hattori, M and Papadimitriou, S. (2007). Berth allocation at indented berths for mega-containerships. European Journal of Operational Research 179. 579–593.

Imai, A, Nishimura, E and Papadimitriou,S. (2008a). Berthing ships at a multi-user container terminal with a limited quay capacity. Transportation Research Part E 44: 136–151.

Imai, A., Chen, H.C., Nishimura, E., Papadimitriou, S., (2008b). The simultaneous berth and quay crane allocation problem. Transportation Research Part E 44 (5), 900–920.

Iris F. A. Vis · Roel G. van Anholt, (2010). Performance analysis of berth configurations at container terminals . OR Spectrum Vol.32.453–476

Iris C., Pacino D. ,Ropke S, Larsen A.,(2105) Integrated Berth Allocation and Quay Crane Assignment Problem: Set partitioning models and computational results. Transportation Research Part E 81 (2015) 75–97

He J, Huang Y,Chang D, Zhang W,(2013). A Knowledge-based System for Berth Allocation in a Container Terminal, , TELKOMNIKA, Vol. 11, No. 5, pp. 2291~ 2300

Dubreuil J. (2008.) La logistique des terminaux portuaires de conteneurs, CIRRELT 2008-38

Ji, M. , Zhu, H.; WangQ.; Zhao R.; Yang.Y. (2105), Integrated strategy for berth allocation and crane assignment on a continuous berth using Monte Carlo simulation, Simulation: Transactions of the Society for Modeling and Simulation International, v 91, n 1, p 26-42, Jan. 2015

Karaboga.D, B Gorkemli, C Ozturk, N Karaboga, (2007), A comprehensive survey: artificial bee colony (ABC) algorithm and applications, Artificial Intelligence Review 42 (1), 21-57

Karaboga, D., Basturk, B (2007). A powerful and effficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm. Journal of Global Optimization, 39(3), 459–471 (2007)

Karam, A. Eltawil, A.B. Functional integration approach for the berth allocation, quay crane assignment and specific quay crane assignment problems. Computers and Industrial Engineering, v 102, p 458-466, December 1, 2016

Kim, K., Moon, K., 2003. Berth scheduling by simulated annealing. Transportation Research B 37, 541–560.

Lalla-Ruiz, E.; Exposito-Izquierdo, C.; de Armas, J.;Melian-Batista, B.;Moreno-Vega, J.M. Migrating Birds Optimization for the Seaside Problems at Maritime Container Terminals, Journal of Applied Mathematics, p 781907 (12 pp.), 2015

Lai, K.K., Shih, K., 1992. A study of container berth allocation. Journal of Advanced Transportation 26, 45–60.

Lee, Y., Chen, C.Y.2009. An optimization heuristic for the berth allocation problem. European Journal of Operational Research. 196.500-508

Liang, C., Huang, Y., Yang, Y., 2008. Research on quay crane scheduling problem by hybrid genetic algorithm. Proceeding in International conference on Automation and Logistics.

Liang, C., Huang, Y., Yang, Y., (2009a). A quay crane dynamic scheduling problem by hybrid evolutionary algorithm for berth allocation planning. Computers and Industrial Engineering. 56 (3), 1021–1028.

Liang, C., Huang, Y., Yang, Y., (2009b). Multiobjective hybrid genetic algorithm for quay crane scheduling in berth allocation planning. Int.J.Manufacturing Technology and

Liang, C., J, Guo, Yang, Y., (2009c). Multiobjective hybrid genetic algorithm for quay dynamic assignment in berth allocation planning. J.Intell. Manuf.

Liang, C., Hwang, H., Gen, M., (2011). A berth allocation planning problem with direct transshipment consideration. J.Intell. Manuf.

Lim, A., (1998). The berth planning problem. Operations Research Letters 22, 105–110.

Ma, H.L. Chan, Felix T.S., Chung, S.H.(2014). A fast approach for the integrated berth allocation and quay crane assignment problem, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, v 229, n 11, p 2076-2087.

Meisel.F. (2009). Seaside Operations Planning in Container Terminals. Contributions to Management science. Physica-Verlag .A Springer Company

Meisel, F., Bierwirth, C., (2005). Integration of berth allocation and crane assignment to improve the resource utilization at a seaport container terminal. In:Haasis, H.-D., Kopfer, H., Schönberger, J. (Eds.), Operations Research Proceedings 2005. Springer, Berlin, pp. 105–110.

Meisel, F., Bierwirth, C., (2009). Heuristics for the integration of crane productivity in the berth allocation problem. Transportation Research Part E 45 (1), 196–209.

Murty K G, , Liu J, Wan Y, Linn R., (2005). A decision support system for operations in a container terminal. Decision Support Systems 39 . 309– 332

.

Nam-Kyu Park, Branislav Dragović,2004. A Study of Container Terminal Planning. OR Spectrum Vol.26.171–192.

Nam,K.C., K.S, Kwak and M.S, Yu.(,2002). Simulation Study of Container Terminal Performance . Journal of Waterway, Port, Coastal and Ocean Engineering

Nishimura, E., Imai, A., Papadimitriou, S., (2001). Berth allocation planning in the public berth system by genetic algorithms. European Journal of Operational Research 131, 282–292.

Park, Y.-M., Kim, K.-H., (2003). A scheduling method for Berth and Quay cranes. Operational Research Spectrum 25, 1–23.

Qing-Mi Hua, Zhi-Hua Hua,b,□, Yuquan Du,2014. Berth and quay-crane allocation problem considering fuel consumption and emissions from vessels. Computers & Industrial Engineering 70 1–10

Raa B., Dullaert W., Schaeren R.V., 2011. An enriched model for the integrated berth allocation and quay crane assignment problem. Expert Systems with Applications .Vol.38 .14136–14147.

Rashidi H.,Tsang E.P.K (2006). Container Terminals: Scheduling Decisions, their Formulation and Solutions. Journal of Scheduling.

Razman Tahar, Khalid H, (2000). Simulation and analysis for the Kelang Container Terminal operations, Logistics Information Management, Vol. 13 Issue: 1, pp.14-20,

Ruiza, José Luis González, Velardeb, Belén Melián-Batistaa,J. Marcos Moreno-Vegaa (2014) Biased random key genetic algorithm for the Tactical Berth Allocation Problem Applied Soft Computing 22 (2014) 60–76

Rodriguez-Molins, Miguel A. Salido, Federico Barber A GRASP-based meta-heuristic for the Berth Allocation Problem and the Quay Crane Assignment Problem by managing vessel cargo holds Appl Intell (2014) 40:273–290

Rodriguez-Molins,M. A. Salido, and F. Barber Robust Scheduling for Berth Allocation and Quay Crane Assignment Problem. Hindawi Publishing Corporation, Mathematical Problems in Engineering. Volume 2014, Article ID 834927, 17 pages

Salhi.A, Alsoufi.G and Yang X, 2017 An evolutionary approach to a combined mixed integer programming model of seaside operations as arise in container ports, Advances in Theoretical and Applied Combinatorial Optimization.

Salido M, Rodriguez-Molins M ,Barber F.,(2011).A decision support system for managing combinatorial problems in container terminals. Knowledge-Based Systems

Steenken, D., Voss, S., Stahlbock, R. (2004). Container terminal operation and operations research A classification and literature review. OR Spectrum, 26, 3-49.

Stahlbock,R, Voß, S. (2008). Operations research at container terminals: a literature update. OR Spectrum, 30. 1–52

Ursavas Evrim, A decision support system for quayside operations in a container terminal Decision Support Systems 59 (2014) 312–324

Vacca I., Matteo Salani, Michel Bierlaire, 2010. Optimization of operations in container terminals: hierarchical vs integrated approaches, Trans-OR

Vis Iris F.A and Koster R., 2003. Transshipment of containers at a container terminal: An overview. European Journal of Operational Research ,147. 1–16.

Wang,F., Lim,A. (2007). Astochastic beam search for the berth allocation problem. Decision Support Systems. 42 (4), 2186-2196.

Yang C, Wang X, Li Z., (2012). An optimization approach for coupling problem of berth allocation and quay crane assignment in container terminal. Computers & Industrial Engineering 63 (2012) 243–253

Yavuz Turkogullari,. B., Taskin, Z. c, Aras, N., & Altinel, I. K. (2016). Optimal berth allocation, time-variant quay crane assignment and scheduling with crane setups in container terminals. European Journal of Operational Research, 254(3), 985–1001.

Yun, Y.S, Choi. (1999). A simulation model for container-terminal operation analysis using an object-oriented approach. Int. J. Production Economics 59 .221-230

Zampelli S., Yannis Vergados, Rowan Van Schaeren,Wout Dullaert, and Birger Raa (2103). The Berth Allocation and Quay Crane Assignment Problem Using a CP Approach

Zhou P.F., Kang H.G., (2008) .Study on Berth and Quay-crane Allocation under Stochastic Environments in Container Terminal. Systems Engineering — Theory & Practice.Vol.28, Issue 1.