A four-decomposition strategies for hierarchically modeling combinatorial optimization problems: framework, conditions and relations

Marouene CHAIEB
LARODEC Institut Supérieur de Gestion de Tunis
41 rue de la liberté, Le Bardo, 2000, Tunisie
Chaieb.marouene@live.fr

Jaber JEMAI
LARODEC Institut Supérieur de Gestion de Tunis
41 rue de la liberté, Le Bardo, 2000, Tunisie
jaber.jemai@ccis.imamu.edu.sa

Khaled MELLOULI
LARODEC Institut Supérieur de Gestion de Tunis
41 rue de la liberté, Le Bardo, 2000, Tunisie
khaled.mellouli@ihec.rnu.tn

***Abstract*—We address the problem of modeling combinatorial optimization problems (COP). COPs are generally complex problems to solve. So a good modeling step is fundamental to make the solution easier. Our approach orients researches to choose the best modeling strategy from the beginning to avoid any problem in the solving process. This paper aims at proposing a new approach dealing with hard COPs particularly when the decomposition process leads to some well-known and canonical optimization sub-problems. We tried to draw a clear framework that will help to model hierarchical optimization problems. The framework will be composed by four decomposition strategies which are: objective based decomposition; constraints based decomposition, semantic decomposition and data partitioning strategy. For each strategy, we present supporting examples from the literature where it was applied. But, not all combinatorial problems can be benefit from the outcomes and benefits of modeling problems hierarchically, rather only particular problems can be modeled like a hierarchical optimization problem. Thus, we propose a set of decomposability conditions for decomposing COPs. Furthermore, we define the types of relationships between obtained sub-problems and how partial solutions can be merged to obtain the final solution.**

Keywords-Combinatorial optimization problems; hierarchiucal optimization; constraint relaxation; semantic decomposition, objective-based decomposition; data-partitioning; parallel processing, sequential processing, gradually mixed optimization, totally mixed optimization.

I. INTRODUCTION

Hierarchical optimization can be viewed as an application of the divide and conquer strategy for handling complex and hard optimization problems. The solution of the main original problem is implicitly determined by a series of optimization problems which must be solved in a predetermined sequence. It consists of decomposing an optimization problem into two or more sub-problems; each sub-problem has its own objectives and constraints. These sub-problems are usually interconnected in a hierarchical structure where a sub-problem in level i coordinates with a sub-problem of level i-1. As mentioned above, hierarchically modeling is a novel technique to facilitate solving complex COPs. A wide range of researches uses this technique to benefit from its advantages which are summarized in the following six benefits: the time minimization, the multidisciplinary, the parallel processing, the reduction of search space, the reusability and the organization. However, not all problems can be modeled hierarchically, but only problems that satisfied at least one decomposability condition. In a next stage of this paper, we will present our major contribution by drawing a clear framework that will help to model hierarchical optimization problems. The proposed framework is composed by four decomposition strategies. First, the decomposition by objectives consists of dividing the principle problem into a set of sub-problems based on objectives. The second strategy is the semantic decomposition which is based on decomposing the initial problem into a set of semantically linked sub-problems. The constraints relaxation strategy consists of relaxing complicated constraints to obtain easier sub-problems to solve. The last strategy is called data-partitioning strategy which consist to divide the initial problem into a set of components based on data characteristics. The set of derived sub-problem should be linked and their partial solution should participate to build the final solution of the main initial problem. That is, we presented the possible relationships that may link two sub-problems within the framework.

The nature of relations between components differs from one decomposition strategy (model) to another. We considered that relations between components of the master problem are divided into two basic categories: dependent sub-problems and independent sub-problems. Each category is divided into two classes. The dependent category is divided into sequential approach and parallel approach. The second category is divided into gradually mixed approach and totally mixed approaches. All these approaches will be detailed in the following sections.

This paper will be organized as follows: the next section will be devoted to present the decomposability conditions framework. In section 3, we will develop the hierarchical decomposition framework. Section 4 will be devoted to detail the relationships between sub-problems in the hierarchical optimization framework by presenting the possible relations between components of the global problem. The paper will then be concluded and some future research perspectives will be addressed in the last section.

II. Decomposability conditions framework

As mentioned above, some conditions shall be verified to model the problem using the hierarchical structure. The four conditions consist to verify if the problem is multi-objective (MOP) or over-constraint (OCP) or multi-decision problem (MDP) and last condition depend on the nature of the data (large scale data sets LSDS or categorical data CD) In the following we detailed each condition separately. (See fig. 1).

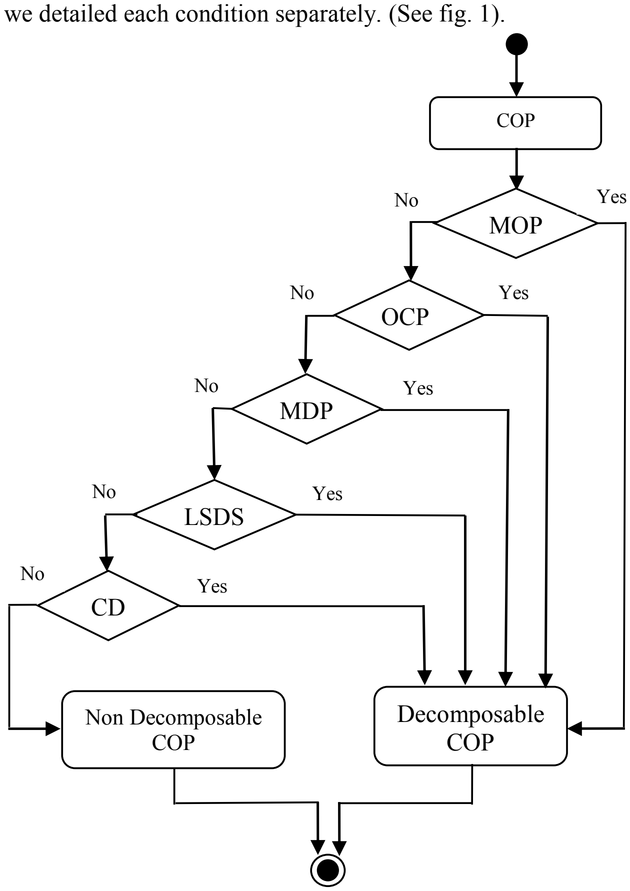

Figure 1. Decomposability conditions framework

A. Multi-objective condition

Considering a particular COP, we have to check if the problem can be decomposed into a set of interconnected components based on targets. If we can split the overall objective into a set of objectives, then we can consider the problem as a multi-objective problem (MOP) and by result we can model it hierarchically.

To illustrate this idea, Mutingi and Mbohwa [1] considered the initial problem as a MOP with three conflicting management goals. The first sub-problem aims at minimizing the schedule cost associated with the trips which is influenced by the nature of the routes assigned to healthcare workers to fulfil the demand requirements. The second component aims at maximizing worker satisfaction which entails meeting the worker preferences to the highest degree possible and especially by ensuring the fairness in workload. The last sub-problem concerns the maximization of client satisfaction which can be expressed as a function of the violations of time windows preferred by the clients. The multi-objective formulation is achieved by optimizing the three objective functions jointly.

B. Over constraint condition

In many real-life applications, most optimization problems are highly constrained where different types of constraints (resources, technical, etc) have to be satisfied in the final solutions. Over constrained problems are complex and difficult to solve like a single monolithic problem. If the set of constraints can be satisfied in different levels (hierarchically), we can conclude that the over constraint condition is satisfied. Thus, the initial complex problem is modeled and solved hierarchically, in each level a set of constraints will be satisfied until gratifying all required (hard) constraints and as much as possible satisfying preferential (soft) constraints. In this context, constraint hierarchies is a new concept proposed to describe high constrained problems by specifying constraints with hierarchical strengths or preferences, i.e required and preferential constraints, most important and less important (preferences) constraints… Moreover, constraint hierarchies allow "relaxing" of constraints with the same strength by applying, e.g., weighted-sum, least-squares or similar comparators.

C. Conditions on data

Two data sets characteristics can help to define if the main problem to solve can be modeled using the proposed hierarchical framework or not. Firstly, if the size of the data set is very large, then it will be possible to divide it into two or more subsets based on a particular criterion (geographic for example). Secondly, in some problems, the data is classified into different types (level of proficiency of nurses, level of patients' illness, etc.). For such problems the data sets can be divided following the defined data types. In the following, we detail each condition

1) *Large scale data sets*

Large scale data sets are collections of so large and complex data inputs that it becomes difficult to process traditionally as one batch. With very large data sets, experiments may face ambiguous situations and may not end. The data decomposition is the other primary form of breaking up monolithic processing into chunks that can be farmed out to multiple cores for parallel processing. The size of the problem space is one of the most obvious candidate measures of complexity which involve modeling a particular problem hierarchically. Problem difficulty was thought to vary with the size of the problem space.

In COOP context, Hertz and Lahrichi [3] presented an illustration of data-partitioning strategy to model and solve the huge size of the problem space of the Home Care Scheduling Problem (HCSP) by decomposing the Canadian territory into 6 districts.

2) *Categorical data*

Another attribute of data sets of a complex problem that can be modeled within the hierarchical framework is the existence of types and categories. Categorical data implicitly divide the input data into classes like types of customers in banking (VIP, Important, Ordinary) or type of employees following their skills (Expert, Skilled, Basic), etc. Such characteristics help to organize the main data set into smaller subsets following the proposed categories which define a set of sub-problems. The main problem will be consequently solved by solving each component and then merging the obtained partial solutions to form the main solution.

To illustrate our approach, let's refer to the previous work [3]. The set of nurses was decomposed into three sets based on level of patients' illness. The first sub-set was named case manager nurses (who typically hold a Bachelor's degree in nursing), which provides care to patient requiring more complex care like coordination of visits and ensuring links with doctors and specialists, organizing the activities of daily living. The second sub-set of nurses was named nurse technicians (who typically hold a community college degree in nursing) which give cares to the short-term clients or long-term clients needing punctual nursing care. The third sub-set of nurses present a surplus team that is not assigned neither any patient nor any district. Their role is to deliver specific nursing care treatments for the client. Authors cited a set of particular intervention of this group of nurses (handle nursing visits that the team nurses are unable to absorb, to absorb visits that are needed outside regular working hours). The last data partitioning decomposition was based on the type of patients which affect the work load of patients because the nurse volume of depends on the time needed to treat his/her specific patients. Thus, authors identified five category of clients based on the type of patients. The first category contains short-term clients that do not require case management (STPCM), the second category contain short-term clients that need post-hospitalization or post-surgery care (STPPH/ STPPPSC), the third category contain long-term clients needing punctual nursing care (LTPPMC), the fourth category contain clients with loss of autonomy (PLA) and the last category contain palliative patients (PP).

D. *Condition 4: Problems with partial nested decisions*

Generally, complex problems are multi-decision problems where some intermediate decisions must be taken to reach a final solution to the main problem. The multi-decisions problems solving process embeds the solving of sub-problems at different times by different decision makers hierarchically at different levels. The final solution will be built then by combining in some way the partial solutions of intermediate sub-problems. Such sub-problems are intuitively easier to handle and to solve than the main problem for different reasons: reduced search space and data sets, uncomplicated combinatorial structure, adapted solving approaches may be already known and solving tools (software) are available. For the above advantages, it is possible to represent multi-decisions problems by its components organized in such a way to fulfil the requirements of the initial problems.

To illustrate by an example, we cite our previous work [4] in which we modeled the HCSP like a hierarchical optimization problem. Our aim was to satisfy a set of conflicted objectives such minimizing travelling costs, maximizing satisfaction level of both patients and nurses... We divided the HCSP into three sub-problems in a hierarchical form based on the order of decisions to take. The three sub-problems defined were the assignment component, the grouping component and the routing component. Before modeling this problem we defined semantically a decision-making map (decision hierarchy) to facilitate the modeling process. We considered that first of all, the assignment task of patients to nurses must be modeled because this first component will affect the decisions made by the latter components. The choice of the next component to model depends on our decision making. Semantically, we think that the grouping component must be defined before the routing component to minimize fuel and by result minimizing costs because the last component depends of the routing component (it is so logic, that after defining the routes to be followed by nurses, sets of nurses with same routes will be travelled in the same groups). May be for other persons thinks that grouping nurses in cluster is more important that the routing component because for authors for example it is important to form multidisciplinary group of nurses than the routing component. Thus the grouping component will be modelled in the second level and the routing component in the last level. (See Fig. 2).

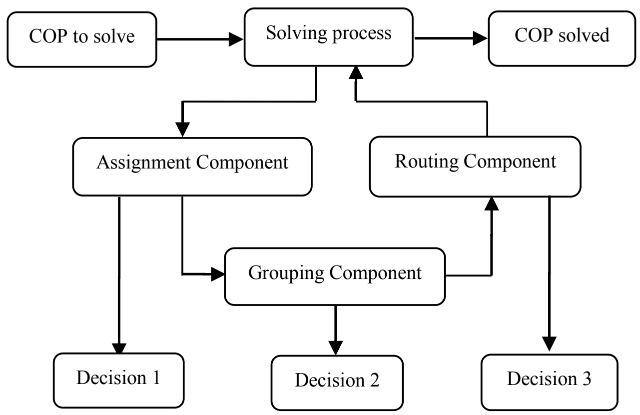

Figure. 2. Example of problems with partial nested decisions

III. PROBLEM DECOMPOSITION STRATEGIES FRAMEWORK

In this section we detail four possible strategies for splitting the main problem into a set of different components, and then we will present some related works to validate the developed framework. In the following a descriptive schema for our developed framework.

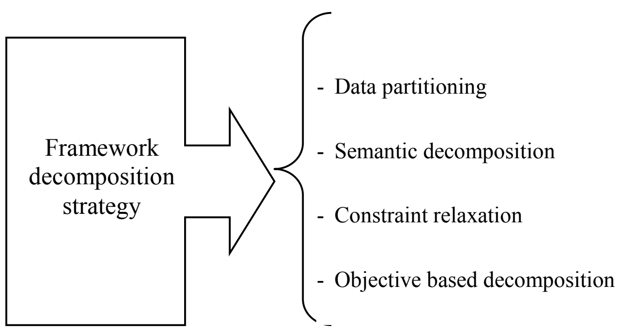

Figure. 3. Hierarchical decomposition framework

A. Objective-based Decomposition

The decomposition by objectives consists of dividing the basic problem into a set of sub-problems based on targets of the initial problem. For each problem at a given level, an optimization sub-problem is formulated to satisfy a set of constraints and achieve a certain objective function. The decomposition by objectives is a very useful and powerful tool to model large scale problems in a hierarchical structure. To develop the proposed strategy, we propose a four steps algorithm which consists to specify first the problem targets, and then propagate such objectives to build their corresponding sub-problems. Then, the components achieve their respective targets and the resulting model meets overall problem targets. Supposing that we have two predefined functions with a particular sub-problem (P) in the input:

- Decomposable(P) determine if P is decomposable or not.
- Solve (P) aims to solve P.

Given a main complex P*i* with an objective O to realize and a set of constraints C to satisfy, and let S*i* the solution of the sub-problem P*i,* the pseudo code is summarized in Algorithm1.

Algorithm1: Objective_Based_Decomposition_Strategy (P)

Input: A main complex problem P {O, C} to solve

Output: A solved problem

BEGIN

Step1: Specify problem (n) targets ($\sum_{i=1}^{n} O_i$*)*

Step2: Assign each target to a sub-problem (P= $\sum_{i=1}^{n} P_i$*)*

Step3: For (i=1; i >=n; i++)

{ If (P_i *is decomposable(* P_i *))*

{Objective_Based_Decomposition_Strategy(P_i *);}*

Else {Solve (Pi);}

Step4: Combining solutions of Pi (S= $\sum_{i=1}^{n} S_i$ *);*

End

In recent years, objective-based decomposition has been extensively applied to deal with large scale problems. An example of this strategy is the Analytical Target Cascading (ATC) that was presented by Kim et al. [5] and used by Ford Motor Company. In ATC, top level targets are propagated to lower levels. The resulting responses are rebalanced at higher levels to achieve consistency. The optimal system solution is obtained through an iterative process until target/response consistency is achieved. Globally, the objective based decomposition strategy has been applied to complex systems such as automotive design in the work of Kim et al. [5] and architectural design in the work of Choudhary et al. [6]. Kokkolaras et al. [7] applied decomposition by objective to model product family. In this context, Mutingi and Mbohwa [1] modeled and developed a robust support tool using the objective based decomposition strategy in which they aims to satisfy three conflicting management goals. The first sub-problem aims to minimize the schedule cost associated with the trips which is influenced by the nature of the routes assigned to healthcare workers to fulfill the demand requirements. The second component aims to maximize worker satisfaction which entails meeting the worker preferences to the highest degree possible and especially by ensuring the fairness in workload. The last sub-problem concerns the maximization of client satisfaction which can be expressed as a function of the violations of time windows preferred by the clients.

B. Semantic Decomposition

The Semantic decomposition strategy aims to isolate embedded sub-problems and their models from the initial problem; solve them and then rebuild the big solution from the obtained partial (intermediate) solutions. It consists of dividing an initial large scale and hard optimization problem into a set of semantically independent sub-problems. Intermediate sub-problems are easier to solve than the initial problem and the quality of their solution affect the quality of the global solution.

For instance, the Machine scheduling problem consists of assigning a set of jobs to a machine in such a way that the capacity constraint is not violated. It is easy to remark that such a constraint is of the same form as that of a knapsack problem. Another important example in which knapsack problems arise is the capital budgeting problem. This problem involves finding a subset of the set of capital projects under consideration that will yield the greatest return on investment, while satisfying specified financial, regulatory and project relationship requirements. The home health care scheduling problem has been also decomposed semantically into three components: assignment problem (P1: Which caregiver will serve which patient?), grouping problem (P2: build the teams that will move together) and a routing problem (P3: design the routes to be followed by the medical teams). In the following, we present an illustrative schema for the modeling of the HCSP using the semantic decomposition strategy.

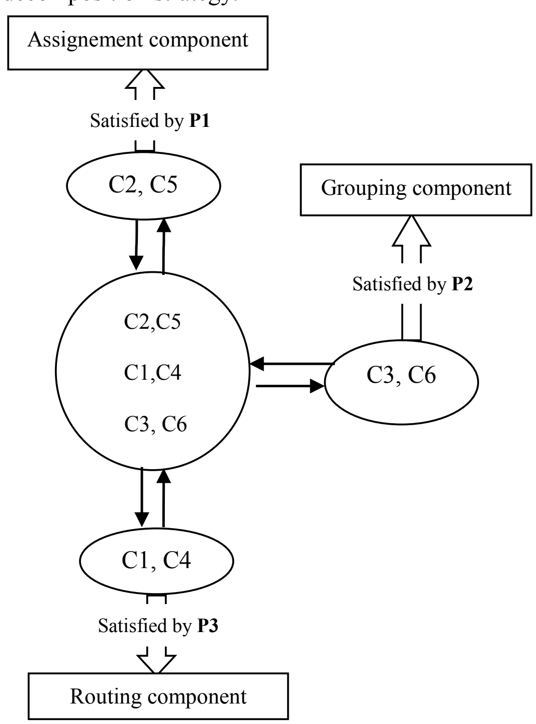

Figure. 4. Illustrative example of the semantic decomposition strategy

C. Constraint Relaxation

Constraints relaxation techniques consists of solving a given optimization problem without considering its whole set of constraints. Some of them will be ignored at a particular point in the solving process. After a while, they will be reintegrated and injected to the current solution. This approach consists in relaxing a set of complicating constraints in order to obtain a more tractable model. By removing the complicating constraints from the constraint set, the resulting sub-problem is frequently considerably easier to solve. This technique requires that one understand the structure of the problem being solved in order to relax the hard constraints.

Many features of constraint relaxation can be detected like relaxing the integrity of decision variables. The bound resulting from a particular decomposition can be computed using many different computational techniques like the Dantzig-Wolfe decomposition in which the solutions to the base problem are generated dynamically and combined in an attempt to obtain a solution satisfying the complicating constraints (Dantzig and Wolfe [8]) and the column Generation techniques (Vanderbeck [9]). The Lagrangian relaxation is another constraint relaxation technique in which the complicating constraints are enforced implicitly by penalizing their violation in the objective function.

A related approach is that of Lagrangian decomposition which consists in isolating sets of constraints so as to obtain multiple, separate, easy-to-solve sub problems. Another constraint relaxation technique is the Decomposition Integer Programming (DIP) in which the fundamental idea is to exploit the ability to either optimize over and/or separate from the convex hull of solutions to a given relaxation in order to derive improved methods of bounding the optimal solution value. We can refer here to some related works to relax the difficulty of the original problem like the works of Ralphs and Galati [10]. The branch and bound is an enumerative algorithm that consists in applying this same method to each of the resulting sub-problems recursively to relax the original problem by branching it. It consists then in finding optimal solutions to various optimization problems after the systematic enumeration of all candidate solutions. It is efficiently used for exact resolution of optimization problems.

Gomory [11] was the first to derive a cutting plane algorithm which is another technique for constraints relaxation in optimization problems. A general cutting plane approach relaxes the integrality restrictions on the variables and solves the resulting LP relaxation. It works by solving a non-integer linear program, the linear relaxation of the given integer program. These two basic methods described above can be hybridized into an algorithm that combines the power of the polyhedral and disjunctive approaches into a single algorithm. This method is called branch and cut. The Branch and cut involves running a branch and bound algorithm and using cutting planes to tighten the linear programming (LP) relaxations. A variety of difficult problems have been solved by reformulating them as either set-covering or set-partitioning problems having an extraordinary number of variables. The column generation is employed to handle with the difficulty to solve directly the set-covering or the set-partitioning problems because of their small instances.

D. Data Partitioning

This strategy consists in partitioning the global problem instance to solve into a collection of data sets to reduce the complexity of the original problem that is usually large scale and difficult to solve in one track. It may be easier to organize the problem as a collection of data sets with well-defined relations rather than attempt to pose a single monolithic problem. The resulting data sets will be resolved iteratively or recursively by applying the same process at different data subsets.

Reviewing the literature, in the field of healthcare, multiple works model the HCSP the data partitioning decomposition strategy. In this context, we cite the work of Eveborn et al. [12] in which they developed a decision support system LAPS CARE to aid planners to solve the staff planning of home care automatically by improving the planning operations and provide better routes for staff.

According to the authors [12], the problem is complex and must be hierarchically decomposed into many partitions using the data-partitioning (recursive) decomposition strategy. Thus they decided to divide the work area into a set of sub-area to make planning easier, where each customer belongs to one sub area and each staff member work in one or several of the sub-areas. So it is important to mention here that all elements of the main problem will become sub-sets of original elements during the local iteration process. If no improvement is done, authors applied a split to the best local to facilitate further progress, so the split process can be viewed as dividing the subsets to even smaller subsets.

Mullinax and Lawley [13] deal with the intensive care nursery provided to ill neonatal infants. According to authors [13] the assignment process is complex and developing balanced nurse workloads is difficult, thus they decided to decompose it into two sub-problems. They decided to divide the data set (infants) into three sub groups based on the patient level which mean the type (state) of the patient. The authors [13] grouped patients into three levels: level I patients that require minimal care; level II patients that require close attention, Level III care for critically ill patients. To classify infants, a neonatal acuity system consists of 14 models contained scores were developed to precise to which level a newborn belong. The data partitioning strategy was also used in the second sub-problem in which the newborn homes were divided into a number of physical zones.

Another work to present in the same research field is the work of Hertz and Lahrichi [3] which presented a perfect illustration of using the data-partitioning decomposition strategy to facilitate the modeling of a complex problem like the HCSP. Given the size of the territory, to balance the work load of nurses while avoiding long travels to visit the clients, authors partitioned the Canadian territory into 6 districts (each one being constituted by several basic units) with each district being assigned to a multidisciplinary team of caregivers. A data partitioning decomposition was also made for both nurses and patients to simplify the initial problem. The set of nurses was divided into three sets based on the level of patients' illness, the first subset was named case manager nurses (who typically hold a Bachelors' degree in nursing) which gives care to patient requiring more complex care such as coordination of visits and ensuring links with doctors and specialists, organizing the activities of daily living. The second subset of patients was named nurse technicians (who typically hold a community college degree in nursing) which give cares to the short-term clients or long-term clients needing punctual nursing care. The third sub-set present a surplus team that is not assigned neither any patient nor any district. Their role is to deliver specific nursing care treatments. The last data partitioning decomposition was based on the type of patients which affect the work load of patients because the nurse volume of depends on the time needed to treat his/her specific patients. Thus, authors identified five categories of clients based on the type of patients. The first category contains short-term clients that do not require case management, the second category contain short-term clients that need post-hospitalization or post-surgery care, the third category contain long-term clients needing punctual nursing care, the fourth category contain clients with loss of autonomy and the last category contain palliative patients.

IV. RELATIONS BETWEEN SUB-PROBLEMS

In this section, we will investigate the possible links and relationships that may appear between sub-problems. That is, to address the how partial solutions of sub-problems can be reintegrated to build a solution for the main problem. There are many ways in which the approximate and accurate representations and solutions can be integrated. The nature of relations between components differs from one decomposition strategy to another.

As mentioned in the introduction of the paper, multiple relations can exist between sub-problems. The following framework present a summary of all possible relations between sub-problems which will be exhaustively detailed and argued by a set of examples in next sections.

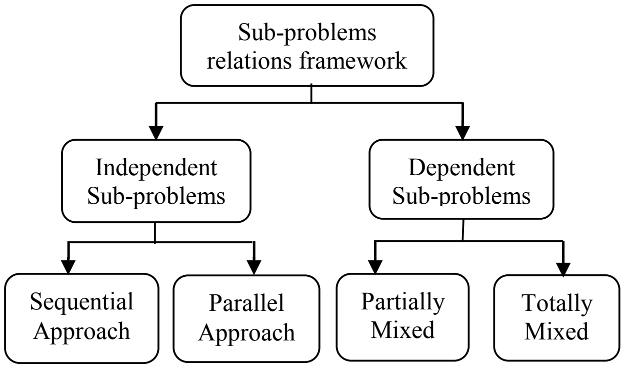

Figure 5. Sub-problems relations framework

A. *Independent Sub-problems Category*

The independent sub-problems category supposes that each sub-problem is executed separately without any mixing with other sub-problems. Reviewing the literature, we remarked that this category was divided into two possible approaches: the sequential approach and the parallel approach. In the following two subsections we detail each approach.

1) *Sequential Approach*

In this approach, the optimization process is started using the least accurate level of representation, then after a certain set number of function evaluations, the optimization on this level is stopped and the results used as starting points for the next more accurate level. This is carried on sequentially and the number of function evaluations is decreased from one level to the next until the most accurate level is reached where fewest function evaluations are carried out.

In this context, El-Beltagy and Kean [14] in which they presented empirical results. Kim et al. [5] named the same approach decomposition method and used it to cooperate between sub-problems in the Analytical Target Cascading.

In the same context, we cite the HCSP in which patients ask caregivers with particular skills to do the required treatments. Once assigned to patients, caregivers will move following specified routes to patients. The home health care problem asks, then, for finding the set of nurses assigned to each patient and also it needs to know the routes to be followed by each team of caregivers to reach its destination. Clearly, each decision comes as an answer to a particular optimization problem. The first problem is an assignment problem where the question is on which caregiver will help which patient. The second problem is on the routes to be followed by vehicles transporting nurses to reach their already assigned patients. Then, the solution of the assignment problem is an input to the routing problem. Consequently, the HCSP can be viewed and modeled as a sequential hierarchical optimization problem. The literature on the HCSP shows three types of studies of the problem: in first class the focus is on the assignment problem, in the second class of papers the main studied part is the routing problem and in some recent papers the problem is handled without omitting or hiding one of its two components.

2) *Parallel Approach*

In this approach, sub-problems of the principle problem interact in a collaborative form. The goal of collaboration is to allow an easy interaction amongst sub-problems on different levels. Thus, the complex problem is hierarchically decomposed into a number of sub-problems which interact by a system-level coordination process. This form of interaction between components is well consulted specifically in case of multidisciplinary environment. Braun and Kroo [15] enumerate some advantageous points of collaborative optimization like reducing the amount of information transferred between disciplines and removing of large iteration-loops. Reviewing the literature, we remarked that this approach has multiple terminologies such as “all at once approach”, “the recursive approach”, “the iterative approach” in which designers talk about a process in which top problem targets are cascaded down to the lowest level in one loop. More details are presented in the work of Kim et al. [5] in which authors used parallel approach in linking different sub-problems.

In this context, we report the problem discussed in the work of Hertz and Lahrichi [3] which present a good illustration of the parallel approach by using a data partitioning strategy in modeling a problem with a very large size (the Canadian territory), which was partitioned into 6 districts {A, B, C, D, E, F}. The six districts will be executed in a parallel structure.

B. *Dependent Sub-problems Category (Mixed Approach)*

As mentioned above, this approach supposes that two or more sub-problems are executed simultaneously. Two types of mixed multilevel optimization approaches have been proposed depending on the scheduling algorithm used to organize the order and the level to reach in the solving of each sub-problem which are the gradually mixed multilevel optimization and the totally mixed optimization. Each approach will be detailed in the next two subsections.

1) *Gradually Mixed Optimization*

The optimization procedure is carried out with multiple levels which are heterogeneously mixed throughout the optimization process. For example, we suppose that our initial problem is composed by three sub-problems. The master problem is gradually mixed optimized if two components are executed simultaneously in a totally mixed form and the third is executed independently. The following figure presents the rate of mixing different levels during the execution process cited in the work of El-Beltagy and Kean [14].

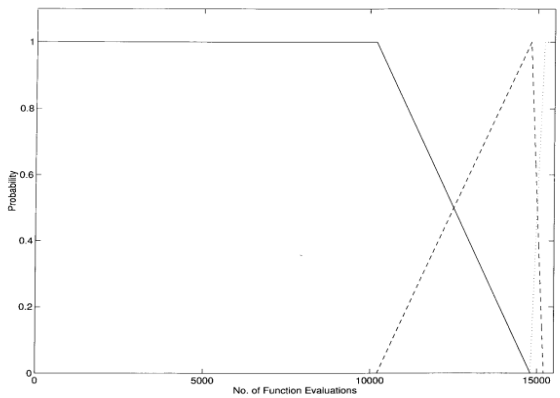

Figure 6. Example of gradually mixed approach

2) Totally Mixed Multilevel Optimization

In totally mixed optimization, the probability of using a particular level is constant throughout most of the optimization process, i.e. all components are executed simultaneously in a totally mixed form. For example, if we suppose that our initial problem is composed of three sub-problems, if the three sub-problems are executed simultaneously in a totally mixed form we say that the initial optimization problem is totally mixed. In the same work, El-Beltagy and Kean [14] gives an example of using the totally mixed approach in which the first level has a probability (y axis) of 82.22 % of the total number of evaluations (x axis), the second has a probability of 16.44 % and the third has a probability of 1.315%.

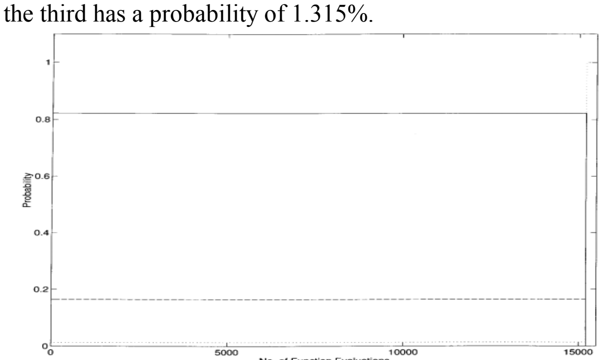

Figure 7. Example of totally mixed approach

V. CONCLUSIONS AND PERSPECTIVES

The hierarchical decomposition frameworks to model complex optimization problems are based on their decomposition into a set of interconnected sub-problems easier to handle. It's an application of the Divide and Conquer strategy to facilitate the handling of difficult problems.

We detailed the necessary conditions to model an optimization problem using the proposed framework. We then presented the four problems decomposition strategies based on objectives, constraints relaxation, data partitioning and semantically. The set of derived sub-problem should be linked and their partial solution should participate to build the final solution of the main initial problem. That is, we presented the possible relationships that may link two sub-problems within the framework. The proposed modeling approach can be applied efficiently to solve many kinds of optimization problems particularly those where the solutions are nested and not trivial to found such the HCSP. In the forthcoming work, we will attempt to validate the proposed framework by modeling the HCSP using the developed framework.

REFERENCES

[1] M. Mutingi, and C. Mbohwa, “A Satisficing Approach to Home Healthcare Worker Scheduling,” International Conference on Law, Entrepreneurship and Industrial Engineering (ICLEIE'2013) Johannesburg South Africa, 2013.

[2] A. R. Clark, H. Walker, “Nurse rescheduling with shift preferences and minimal disruption,” Journal of Applied Operational Research, vol. 3, no. 3, 2011, pp. 148-162.

[3] A. Hertz, and N. Lahrichi, “A patient assignment algorithm for home care services,” Journal of the Operational Research Society, 2009, vol. 60, pp. 481-495.

[4] J. Jemai, M. Chaieb and K. Mellouli, “The home care scheduling problem: A modeling and solving issue, in Proc. Of the 5th International Conference on Modeling, Simulation and Applied Optimization, Tunisia, Hamamet. 2013.

[5] H. Kim, N. Michelena, P. Papalambros, and T. Jiang, “Target cascading in optimal system design,” Journal of Mechanical Design, vol, 125, no. 3, 2002, pp. 474-480.

[6] R. Choudhary, A. Malkawi, and P. Papalambros, “Analytic target cascading in simulation-based building design,” Automation in Construction, vol. 14, no. 4, 2005, pp. 551-568.

[7] M. Kokkolaras, R. Fellini, H. Kim, N. Michelena, and P. Papalambros, “Target cascading in vehicle redesign: a class vi truck study,” International Journal of Vehicle Design, vol. 24, no. 4, 2002, pp. 293-301.

[8] G. Dantzig, and P. Wolfe, “Decomposition principle for linear programs,” Operations Research, vol. 8, 1960, pp. 101-111.

[9] F. Vanderbeck, “On dantzig-wolfe decomposition in integer programming and ways to perform branching in a branch-and-price algorithm,” Operations Research, vol. 48, 2000, pp. 111-128.

[10] T. Ralphs, and M. Galati, editors. Decomposition in integer programming. CRC Press, 2005.

[11] R. E. Gomory, “Outline of an algorithm for integer solutions to linear programs,” Bulletin of the American Mathematical Monthly, vol. 64, 1958, pp. 275-278.

[12] P. Eveborn, P. Flisbergb, and M. Rnnqvist, “Laps carean operational system for staff planning of home care,” European Journal of Operational Research, vol. 177, no. 3, 2006, pp. 962-976.

[13] C. Mullinax, and M. Lawley, “Assigning patients to nurses in neonatal intensive care,” Journal of the operational research society, vol. 53, 2006, pp. 25-35.

[14] M. El-Beltagy, and A. Kean, “A comparison of various optimization algorithms on a multilevel problem,” Engineering Applications of Artificial Intelligence, vol. 12, no. 8, 1999, pp. 639-654.

[15] R. Braun, and I. Kroo, "Development and application of the collaborative optimization architecture in a multidisciplinary design environment multidisciplinary design optimization: State of the art,” SIAM, 1995, pp. 98-116.