Journal of Applied Science and Agriculture, 9(1) January 2014, Pages: 11-25

AENSI Journals

Journal of Applied Science and Agriculture

ISSN 1816-9112

Journal home page: www.aensiweb.com/jasa/index.html

A Review on Decomposition Methods in Engineering Problems

[1]Mehrdad Beykverdi, [2]Azar Fakharian, [3]M. Jazaeri

[1]*Department of Electrical Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.*
[2]*Department of Electrical Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.*
[3]*Electrical and Computer Engineering Faculty, Semnan University, Semnan, Iran.*

ARTICLE INFO

Article history:
Received 16 November 2013
Received in revised form 18 December 2013
Accepted 22 December 2013
Available online 5 March 2014

Keywords:
Decomposition, Benders, Dantzing, Constraints, Objective function, Primal, Dual, Decoupling.

ABSTRACT

Decomposition refers to the strategy of breaking up a large, difficult-to-solve problem into two or more smaller, easier-to-solve problems, such that the solution to the decomposed problems can be used to obtain the solution to the original problem. In the last years, decomposition techniques such as Benders decomposition are used efficiently and extensively to solve large scale optimization problems (particularly, in mixed integer variable optimization problems such as unit commitment and short term hydrothermal coordination). This paper reviews two basic decomposition methods (primal and dual decomposition) and used them to construct Dantzing-wolfe and Benders decomposition techniques.

To Cite This Article: Mehrdad Beykverdi, Azar Fakharian, M. Jazaeri., A Review on Decomposition Methods in Engineering Problems. ***J. Appl. Sci. & Agric.,*** *9(1): 11-25, 2014*

INTRODUCTION

Decomposition is a general approach to solving a problem by breaking it up into smaller ones and solving each of the smaller ones separately, either in parallel or sequentially.

There are few decomposition methods to solve multi-variable problems. The famous decomposition methods are primal, dual (Pennanen, H., *et al*., 2011; Torresani, L., *et al*., 2013), Dantzing-Wolfe (Aganagic, M., S. Mokhtari, 1997) and Benders decomposition (Sifuentes, W.S., A. Vargas, 2007; Hoang Hai Hoc, 1982; Stephen Boyd, Lieven Vandenberghe, 2009). In this paper, these four famous decomposition methods will be described. In the next section, primal decomposition without constraints is introduced. The theory of dual decomposition without constraints is presented in section 3. Decomposition with constraints is discussed in section 4. The formulation of Dantzing-Wolfe decomposition is explained in section 5. In section 6, we elaborate Benders decomposition and an example solution with it. In section 7, decoupling method and its difference with decomposition process is described. Finally, in the last section a case study according to Benders method is presented and the main conclusions of the paper are summarized.

Primal Decomposition without Constraints:

At outset, we'll consider the simplest possible case, an unconstrained optimization problem that splits into two sub- problem. (But note that the most impressive applications of decomposition occur when the problem is split into many sub-problems.) In our first perusal, we consider an unconstrained minimization problem, of the form

$$Minimize\ f(x) = f_1(x_1, y) + f_2(x_2, y) \tag{1}$$

Where the variable is: $x = (x_1, x_2, y)$. Although the dimensions don't matter here, it's useful to think of x_1 and x_2 as having relatively high dimension, and y having relatively small dimension. The objective is almost block separable in x_2 ; indeed, if we fix the sub-vector y , the problem becomes separable x_1 and x_2 , and therefore can be solved by solving the two sub-problem independently. For this reason, y is called the complicating variable, because when it is fixed, the problem splits or decomposes. In other words, the variable y

Corresponding Author: Mehrdad Beykverdi, Department of Electrical Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.
E-mail: Mehrdad_b2003@yahoo.com & +98- 9124570855

complicates the problem. It is the variable that couples the two sub-problems. It considers $x_1(x_2)$ as the private variable or local variable associated with the first (second) sub-problem, and y as the public variable or interface variable or boundary variable between the two sub-problems.

The observation that the problem becomes separable when y is fixed suggests a method for solving the problem (1). Let $\varphi_1(y)$ denotes the optimal value of the problem and similarly, let $\phi_2(y)$ denotes the optimal value of the problem (Yong Fu, M. Shahidehpour, Zuyi Li, 2005):

$$Minimize\ e_{x1} = f_1(x_1, y) \tag{2}$$

$$Minimize\ e_{x2} = f_2(x_2, y) \tag{3}$$

(Note that if f_1 and f_2 are convex, so are φ_1 and φ_2 .) We refer to (2) as sub-problem 1, and (3) as sub-problem 2. Then the original problem (1) is equivalent to the problem.

$$Minimize\ e_y = \varphi_1(y) + \varphi_2(y) \tag{4}$$

This problem is called the master problem. If the original problem is convex, so is the master problem. The variables of the master problem are the complicating or coupling variables of the original problem. The objective of the master problem is the sum of the optimal values of the sub-problem.

A decomposition method solves the problem (1) by solving the master problem, using an iterative method such as the sub-gradient method. Each iteration requires solving the two sub-problem in order to evaluate $\varphi_1(y)$ and $\varphi_2(y)$ and their gradients or sub-gradients. This can be done in parallel, but even if it is done sequentially, there will be substantial savings if the computational complexity of the problems grows more than linearly with problem size.

Let's see how to evaluate a sub-gradient of φ_1 in y , assuming the problem is convex. We first solve the associated sub-problem, i.e., we find $\underline{x}_1(y)$ that minimizes $f_1(x_1, y)$. Thus, there is a sub-gradient of f_1 of the form $(0, g_1)$, and not surprisingly, g_1 is a sub-gradient of φ_1 in y . We can carry out the same procedure to find a sub-gradient $g_2 \in \partial\varphi_2(y)$, and then $g_1 + g_2$ is a sub-gradient of $\varphi_1 + \varphi_2$ in y .

We can solve the master problem by a variety of methods, including bisection (if the dimension of y is one), gradient or quasi-Newton methods (if the functions are differentiable), or sub-gradient, cutting-plane, or ellipsoid methods (if the functions are non-differentiable).

This basic decomposition method is called primal decomposition because the master algorithm manipulates (some of the) primal variables.

When we use a sub-gradient method to solve the master problem, we get a very simple primal decomposition algorithm. Repeat below stages:

- Solve the sub-problem (possibly in parallel).
- Find $\underline{x}_1$ that minimized $f_1(x_1, y)$, and a sub-gradient $g_1 \in \partial\varphi_1(y)$.
- Find $\underline{x}_2$ that minimized $f_2(x_2, y)$, and a sub-gradient $g_2 \in \partial\varphi_2(y)$.
- Update complicating variable.

$$y = y - \alpha_k(g_1 + g_2) \tag{5}$$

Here α_k is a step length that can be chosen in any of the standard ways.

We can interpret this decomposition method as follows. We have two sub-problem, with private variables or local variables x_1 and x_2 , respectively. We also have the complicating variable y which appears in both sub-problems. At each step of the master algorithm the complicating variable is fixed, which allows the two sub-problems to be solved independently. From the two local solutions, we construct a sub-gradient for the master problem, and using this, we update the complicating variable. Then we repeat the process. When a sub-gradient method is used for the master problem, and φ_1 and φ_2 are differentiable, the update has a very simple interpretation. We interpret g_1 and g_2 as the gradients of the optimal value of the sub-problem, with respect to the complicating variable y . The update simply moves the complicating variable in a direction of improvement of the overall objective.

The primal decomposition method works well when there are few complicating variables, and we have some good or fast methods for solving the sub-problem. For example, if one of the sub-problem is quadratic, we can solve it analytically; in this case the optimal value is also quadratic, and given by a Scour complement of the local quadratic cost function. (But this trick is so simple that most people would not call it decomposition.)

The basic primal decomposition method described above can be extended in several ways. We can add separable constraints, i.e., constraints of the form $x_1 \in C_1$ and $x_2 \in C_2$. In this case (and also, in the case when dome φ is not all vectors) we have the possibility that $\varphi_i(y) = \infty$ (i.e., $y \notin dom\ \varphi$) for some choices of y . In this case we find a cutting-plane that separates y from domain φ , to use in the master algorithm (Shahidehpour, M., Yong Fu, 2005).

Dual Decomposition Without Constraints:

We can apply decomposition to the problem (1) after introducing some new variables, and working with the dual problem. We first express the problem as by introducing a new variable and equality constraint:

$$\begin{aligned} &Minimize\ f(x) = f_1(x_1, y_1) + f_2(x_2, y_2) \\ &subject\ to\ \ y_1 = y_2 \end{aligned} \tag{6}$$

A local version of the complicating variable y, along with a consistency constraint that requires the two local versions to be equal is introduced. Note that the objective is now separable, with the variable partition (x_1, y_1) and (x_2, y_2). Now we form the dual problem. The Lagrangian is separable (Bo Lu, M. Shahidehpour, 2005).

$$L(x_1, y_1, x_2, y_2, \upsilon) = f_1(x_1, y_1) + f_2(x_2, y_2) + \upsilon^T y_1 - \upsilon^T y_2 \tag{7}$$

The dual function is:

$$g(\upsilon) = g_1(\upsilon) + g_2(\upsilon) \tag{8}$$

Where:

$$g_1(\upsilon) = \inf_{x_1, y_1}\left(f_1(x_1, y_1) + +\upsilon^T y_1\right) \tag{9}$$

$$g_2(\upsilon) = \inf_{x_2, y_2}\left(f_2(x_2, y_2) + +\upsilon^T y_2\right) \tag{10}$$

Note that g_1 and g_2 can be evaluated completely independently, e.g., in parallel. Also note that g_1 and g_2 can be expressed in terms of the conjugates of f_1 and f_2 :

$$g_1(\upsilon) = -f_1^*(0, -\upsilon), \quad g_2(\upsilon) = -f_2^*(0, -\upsilon), \tag{11}$$

The dual problem with υ variable is:

$$Maximize : g_1(\upsilon) + g_2(\upsilon) = -f_1^*(0, -\upsilon) - f_2^*(0, -\upsilon) \tag{12}$$

This is the master problem in dual decomposition. The master algorithm solves this problem using a sub-gradient, cutting-plane, or other method. To evaluate a sub-gradient of $-g_1$ (or $-g_2$) is easy. We find $\underline{x}_1$ and $\underline{y}_1$ that minimize $f_1(x_1, y_1) + +\upsilon^T y_1$ over x_1 land y_1. Then a sub-gradient of $-g_1$ at υ is given by $-\underline{y}_1$. Similarly, if $\underline{x}_2$ and $\underline{y}_2$ minimize $f_2(x_2, y_2) - \upsilon^T y_2$ over x_2 and y_2, then a sub-gradient of $-g_2$ at υ is

Journal of Applied Science and Agriculture, 9(1) January 2014, Pages: 11-25

given by $-\underline{y}_2$. Thus, a sub-gradient of the negative dual function $-g$ is given by $\underline{y}_2 - \underline{y}_1$, which is nothing more than the consistency constraint residual.

If we use a sub-gradient method to solve the master problem, the dual decomposition algorithm has a very simple form.

Repeat to solve the sub-problem (possibly in parallel):

- Find x_1 and y_1 that minimize $f_1(x_1, y_1) + +\upsilon^T y_1$.
- Find x_2 and y_2 that minimize $f_2(x_2, y\ ) - \upsilon^T y_2$.
- Update dual variables (prices).

$$\upsilon := \upsilon - \alpha_k (y_2 + y_1) \tag{13}$$

$$\upsilon = \upsilon - \alpha_k (y_2 - y_1) \tag{14}$$

Here α_k is a step size which can be chosen several ways. If the dual function g is differentiable, then we can choose a constant step size, provided it is small enough. Another choice in this case is to carry out a line search on the dual objective. If the dual function is non-differentiable, we can use a diminishing non-sum able step size, such as $\alpha_k = \alpha / k$.

At each step of the dual decomposition algorithm, we have a lower bound on p^*, the optimal value of the original problem, given by:

$$p^* \geq g(\upsilon) = f_1(x_1, y_1) + \upsilon^T y_1 + f_2(x_2, y_2) - \upsilon^T y_2 \tag{15}$$

That x_1, y_1, x_2, y_2 is the iterations. Generally, the iterations are not feasible for the original problem, i.e., we have $y_2 - y_1 \neq 0$. (If they are feasible, we have maximized *g*.) A reasonable guess of a feasible point can be constructed from this iterate as: $(x_1, y_1), (x_2, y_2)$ where:

$$y = (y_1 + y_2)/2 \quad \underline{y} = (y_1 + y_2)/2 \tag{16}$$

In other words, y_1 and y_2 (which are different) is replaced with their average value. (The average is the projection of (y_1, y_2) onto the feasible set $(y_1 = y_2)$.) This gives an upper bound on p^*, given by:

$$p^* \leq f_1(x_1, y_1) + f_2(x_2, y_2) \tag{17}$$

A better feasible point can be found by replacing y_1 and y_2 with their average, and then solving the two sub-problems (2) and (3) encountered in primal decomposition, i.e., by evaluating $\varphi_1(y) + \varphi_2(y)$. This gives the bound:

$$p^* \leq \varphi_1(y) + \varphi_2(y) \tag{18}$$

Dual decomposition has an interesting economic interpretation. We imagine two separate economic units, each with its own private variables and cost function, but also with some coupled variables. We can think of y_1 as the amounts of some resources consumed by the first unit, and y_2 as the amounts of some resources generated by the second unit. Then, the consistency condition $y_1 = y_2$, means that supply is equal to demand. In primal decomposition, the master algorithm simply fixes the amount of resources to be transferred from one unit to the other, and updates these fixed transfer amounts until the total cost is minimized.

In dual decomposition, υ as a set of prices for the resources is interpreted. The master algorithm sets the prices, not the actual amount of the transfer from one unit to the other. Then, each unit independently operates in such a way that its cost, including the cost of the resource transfer (or profit generated from it), is minimized. The dual decomposition master algorithm adjusts the prices in order to bring the supply into consistency with the demand. In economics, the master algorithm is called a price adjustment algorithm, or atonement procedure (Zuyi Li, M. Shahidehpour, 2005).

There is one subtlety in dual decomposition. Even if we do find the optimal prices, υ^*, there is the question of finding the optimal values of x_1, x_2 and y. When f_1 and f_2 are strictly convex, the points found in evaluating g_1 and g_2 are guaranteed to converge to optimal, but in general the situation can be more difficult. There are also some standard tricks for regularizing the sub-problem that work very well in practice. As in the primal decomposition method, it can be encountered infinite values for the sub-problem. In dual decomposition, it has $g_i(\upsilon) = -\infty$, this can occur for some values of υ, if the functions f_i grow only linearly in y_i.

Decomposition with Constraints:

So far, it has been considered the case, where two problems would be separable, except for some complicating variables that appear in both sub-problems. We can also consider the case where the two sub-problems are coupled via constraints that involve both sets of variables. As a simple example, suppose our problem has the form:

$$\begin{aligned} &Minimize\ f_1(x_1) + f_2(x_2) \\ &subject\ to\ x_1 \in C_1, x_2 \in C_2 \\ &\qquad\qquad h_1(x_1) + h_1(x_1) \le 0 \end{aligned} \tag{19}$$

Here C_1 and C_2, are the feasible sets of the sub-problem, presumably described by linear equalities and convex inequalities. The functions $h_1 : R^n \to R^p$ and $h_2 : R^n \to R^p$ have components that are convex. The sub-problem are coupled via the p constraints that involve both x_1 and x_2. We refer to these as complicating constraints (since without them, the problems involving x_1 and x_2 can be solved separately) (Yong Fu, M. Shahidehpour, Zuyi Li, 2006).

Primal Decomposition:

To use primal decomposition, we can introduce a variable $t \in R^p$ that represents the amount of the resources allocated to the first sub-problem. As a result, $-t$ is allocated to the second sub-problem. The first sub-problem becomes:

$$\begin{aligned} &Minimize\ f_1(x_1) \\ &subject\ to\ x_1 \in C_1,\ h_1(x_1) \le t \end{aligned} \tag{20}$$

And the second one:

$$\begin{aligned} &Minimize\ f_2(x_2) \\ &subject\ to\ x_2 \in C_2,\ h_2(x_2) \le t \end{aligned} \tag{21}$$

Let $\varphi_1(t)$ and $\varphi_2(t)$ denote the optimal values of the sub-problem (20) and (21), respectively. Evidently the original problem (19) is equivalent to the master problem of minimizing $\varphi(t) = \varphi_1(t) + \varphi_2(t)$ over the allocation vector t. These sub-problem can be solved separately, when t is fixed. Not surprisingly, we can find a sub-gradient for the optimal value of each sub-problem from an optimal dual variable associated with the coupling constraint. Let $p(z)$ be the optimal value of the convex optimization problem:

$$\begin{aligned} &Minimize\ f(x) \\ &subject\ to\ x \in X,\ h(x) \le z \end{aligned} \tag{22}$$

And suppose $z \in dom\ p$. Let $\lambda(z)$ be an optimal dual variable associated with the constraint: $h(x) \le z$ then, $-\lambda(z)$ is a sub-gradient of p in z. To see this, we consider the value of p at another point $\breve{z}$:

$$\begin{aligned}
p(\breve{z}) &= \sup_{\lambda\geq 0}\inf_{x\in X}\left(f(x)+\lambda^T(h(x)-\breve{z})\right)\geq \inf_{x\in X}\left(f(x)+\lambda^T(z)^T h(x)-\breve{z}\right)\\
&= \inf_{x\in X}\left(f(x)+\lambda(z)^T(h(x)-z+z-\breve{z})\right)\\
&= \inf_{x\in X}\left(f(x)+\lambda(z)^T(h(x)-z)\right)+\lambda(z)^T(z-\breve{z})\\
&= \varphi(z)+(-\lambda(z))^T(\breve{z}-z)
\end{aligned} \tag{23}$$

This holds for all points $\breve{z}\in dom\ p$, so $-\lambda(z)$ is a sub-gradient of p in z.

Thus, to find a sub-gradient of φ, we solve the two sub-problem, to find optimal x_1 and x_2, as well as optimal dual variables λ_1 and λ_2 associated with the constraints $h_1(x_1)\leq t$ and $h_2(x_2)\leq -t$, respectively. Then we have:

$$\lambda_2-\lambda_1\in\partial\varphi(t) \tag{24}$$

It is also possible that $t\notin dom\varphi$. In this case we can generate a cutting plane that separates t from dome φ, for use in the master algorithm. Primal decomposition, using a sub-gradient master algorithm, has the following simple form (Jae Hyung Roh, M. Shahidehpour, Yong Fu, 2007):

- Solve the sub-problem (possibly in parallel).
- Solve (8), to find an optimal x_1 and λ_1.
- Solve (9), to find an optimal x_2 and λ_2.
- Update resource allocation.

$$t:=t-\alpha_k(\lambda_2-\lambda_1) \tag{25}$$

Here α_k is an appropriate step size. At every step of this algorithm we have points that are feasible for the original problem.

Dual Decomposition:

Dual decomposition for this case is straightforward. We form the partial Lagrangian which is separable:

$$\begin{aligned}
L(x_1,x_2,\lambda) &= f_1(x_1)+f_2(x_2)+\lambda^T(h_1(x_1)+h_2(x_2))\\
&= \left(f_1(x_1)+\lambda^T h_1(x_1)\right)+\left(f_2(x_2)+\lambda^T h_2(x_2)\right)
\end{aligned} \tag{26}$$

So it can be minimized over x_1 and x_2 separately, given the dual variable λ, to find:

$$g(\lambda)=g_1(\lambda)+g_2(\lambda) \tag{27}$$

For example, to find $g_1(\lambda)$, the sub-problem below should be solved:

$$\begin{aligned}
&Minimize\ \ f_1(x_1)+\lambda^T h_1(x_1)\\
&subject\ to\ \ x_1\in C_1
\end{aligned} \tag{28}$$

And to find $g_2(\lambda)$, this sub-problem should be solved:

$$\begin{aligned}
&Minimize\ \ f_2(x_2)+\lambda^T h_2(x_2)\\
&subject\ to\ \ x_2\in C_2
\end{aligned} \tag{29}$$

A sub-gradient of $-g_1$ at λ is, naturally, $h_1(x_1)$, $\underline{x}_1$ is any solution of sub-problem (10). To find a sub-gradient of g, the master problem objective, we solve both sub-problem, to get solutions $\underline{x}_1$ and $\underline{x}_2$, respectively. A sub-gradient of $-g$ is then $h_1(\underline{x}_1)+h_2(\underline{x}_2)$. The master algorithm updates (the price vector) λ based on this sub-gradient. If we use a projected sub-gradient method to update λ we get a very simple algorithm. Repeat below stages:

- Solve the sub-problem (possibly in parallel).
- Solve (10) to find an optimal $\underline{x}_1$.
- Solve (11) to find an optimal $\underline{x}_2$.
- Update dual variables (prices).

$$\lambda:=\left(\lambda+\alpha_k\left(h_1(\underline{x}_1)+h_2(\underline{x}_2)\right)\right) \tag{30}$$

At each step we have a lower bound on p^*, given by:

$$g(\lambda) = g_1(\lambda) + g_2(\lambda) = f_1(x_1) + \lambda^T h_1(x_1) + f_2(x_2) + \lambda^T h_2(x_2) \tag{31}$$

The iterations in the dual decomposition method need not be feasible.

$$h_1(x_1) + h_2(x_2) \le 0 \tag{32}$$

At the cost of solving two additional sub-problems, however, we can (often) construct a feasible set of variables, which will give us an upper bound on p^*. When:

$$h_1(x_1) + h_2(x_2) \le 0 \tag{33}$$

It's defined:

$$t = (h_1(x_1) + h_2(x_2))/2 \tag{34}$$

And solve the primal sub-problem (20) and (21). This is nothing more than projecting the current (infeasible) resources used, $h_1(x_1)$ and $h_2(x_2)$, onto the set of feasible resource allocations, which must sum to no more than 0. As in primal decomposition, it can happen that $t \notin dom\,\varphi$. But when $t \notin dom\,\varphi$, this method gives a feasible point and an upper bound on p^* (Jae Hyung Roh, M. Shahidehpour, Yong Fu, 2007).

Coupling constraints and coupling variables:

Except for the details of computing the relevant sub-gradients, primal and dual decomposition for problems with coupling variables and coupling constraints seem quite similar. In fact, we can readily transform each into the other. For example, we can start with the problem with coupling constraints (19), and introduce new variables y_1 and y_2, that bound the subsystem coupling constraint functions, to obtain

$$\begin{aligned} &Minimize\ f_1(x_1)\, f_2(x_2) \\ &subject\ to\ x_1 \in C_1, h_1(x_1) \le y_1 \\ &\qquad x_2 \in C_2, h_2(x_2) \le -y_2 \\ &\qquad y_1 = y_2 \end{aligned} \tag{35}$$

We now have a problem of the form (17), i.e., a problem that is separable, except for a consistency constraint that requires two (vector) variables of the sub-problem to be equal. Any problem that can be decomposed into two sub-problem that are coupled by some common variables, or equality or inequality constraints, can be put in this standard form, i.e., two sub-problem that are independent except for one consistency constraint that requires a sub variable of one to be equal to a sub-variable of the other. Primal or dual decomposition is then readily applied; only the details of computing the needed sub-gradients for the master problem vary from problem to problem (Yong Fu, M. Shahidehpour, 2007).

Dantzing-Wolfe Decomposition:

Another important decomposition technique is *Dantzing-Wolfe decomposition* developed by Dantzing and Wolfe. Dantzing-Wolfe decomposition is related to Benders decomposition in that it is equivalent to performing Benders decomposition on the dual of some linear program. As Benders decomposition is an iterative procedure in which a new row is added to the master program after any iteration, Dantzing-Wolfe decomposition is an iterative procedure in which a new column is added to the master program after any iteration. Dantzing-Wolfe decomposition can be applied to problems with block angular structure. The basic idea of this method is to solve the problems of the following form:

$$\begin{aligned} &Minimize\, Z = C_1^T X_1 + C_2^T X_2 \\ &subject\ to\ A_1 X_1 + A_2 X_2 = b \\ &\qquad B_1 X_1 = b_1 \\ &\qquad B_1 X_2 = b_2 \\ &\qquad X_1 \ge 0, X_2 \ge 0 \end{aligned} \tag{36}$$

Where A_1, A_2, B_1 and B_2 are matrices with sizes $m\times n_1$, $m\times n_2$, $m_1\times n_1$ and $m_2\times n_2$, respectively. Furthermore, c_1, c_2, X_1 and X_2 are vectors of sizes n1 ×1, n2 ×1, n1 ×1 and n1 ×1, respectively. For the sake of simplicity, it is assumed S_1 and S_2 are bounded:

$$S_1 = \{X_1 | X_1 \ge 0, B_1 X_1 = b_1\} \tag{37}$$

Journal of Applied Science and Agriculture, 9(1) January 2014, Pages: 11-25

$$S_2 = \{X_2 | X_2 \geq 0, B_2 X_2 = b_2\} \tag{38}$$

Let $W_1 = \{X_{11}, X_{12}, ..., X_1 K_1\}$ and $W_2 = \{X_{21}, X_{22}, ..., X_2 K_2\}$ be the sets of all extreme points of the convex polyhedron S_1 and $_2$, respectively. So, any point x ϵ S_j can be written as:

$$\mathrm{x} = \sum_{\mathrm{k}=1}^{\mathrm{kj}} \lambda_{jk} . x_{jk}, where \sum_{k} \lambda_{jk} = 1, \lambda_{jk} = 0 \ (k = 1,2; K; 1,2)$$

$$p_{jk} = A_j . X_{jk} \tag{39}$$

$$f_{jk} = C_j^T . X_{jk}$$

$$for\, k = 1,2,..., K_j \ and \ j = 1,2.$$

Then, the following full master program is equivalent to:

$$Minimize \sum_{k=1}^{k_1} f_{1k} . \lambda_{1k} + \sum_{k=1}^{k_2} f_{2k} . \lambda_{2k}$$

$$subject\,to \sum_{k=1}^{k_1} p_{1k} . \lambda_{1k} + \sum_{k=1}^{k_2} p_{2k} . \lambda_{2k}$$

$$\sum_{k=1}^{k_1} \lambda_{1k} = 1 \tag{40}$$

$$\sum_{k=1}^{k_2} \lambda_{2k} = 1$$

$$\lambda_{1k} \geq 0 \quad k = 1,2,...,.k_1$$

$$\lambda_{2k} \geq 0 \quad k = 1,2,...,.k_1$$

This master program is completely equivalent to the original. It has only $m+2$ rows, compared to the $m + m_1 + m_2$ rows of the original problem. It also has as many columns as the sum of the numbers of extreme points of polyhedrons S_1 and S_2, i.e., $k_1 + k_2$ columns. To solve this full master program, simplex method can be used. Simple method basically looks to the extreme points of the feasible set one by one to check whether it is the optimal solution. However, in our case, the number of extreme points ($k_1 + k_2$) may be very large (Jianhui Wang, M. Shahidehpour, Zuyi Li, 2008). Instead of checking all these points, a technique called *column generation* is used. In this technique, rather than tabulating all columns, columns are created to enter the basis as they are needed. Since the master program has $m+2$ equation constraints, a feasible basis will consist of $m+2$ columns, that is, these columns are linearly independent, and the unique solution of the constraint equations obtained by setting to zero those variables associated with all other columns is nonnegative. If the simple method is used in performing the calculations, there will also be the $m+2$ vector of prices ($\pi, \hat{\pi}$) the *m*-vector π associated with the first m constraints and the 2-vector $\pi = (\pi^1, \pi^2)$ with the remaining two. The inner product of the price vector with any column of the basis must be equal to the cost associated with that column; in the case of master program this relation can be written as:

$$\pi P_{1k} + \pi^1 = f_{1k} \tag{41}$$

$$\pi P_{2k} + \pi^2 = f_{2k} \tag{42}$$

One step of the simple method iteration for solving the master program would be performed as follows: find a column of the constraint matrix whose reduced cost is negative, that is, for which:

$$f_{jk} - \pi P_{jk} - \hat{\pi}_j < 0 \tag{43}$$

Where j=1, 2. Add this column to the current basis, and delete one column from the basis in such a way that the new basis is still feasible. If no column satisfying the above inequality can be found, then the current solution solves the master problem. Otherwise, the simple method gives the appropriate rules for the removal of a column from the basis and for the calculation of the new prices associated with the new basis, with which the next iteration step can begin.

Dantzing-Wolfe Decomposition Algorithm:

Step 1: Assume that an initial basic feasible solution for the master program is available, with basis matrix B and simplex multipliers (price vectors) $(\pi, \hat{\pi})$.Using the simplex multipliers, solve the following sub-problem:

$$\begin{aligned} &\textit{Minimize } z_j = (c_j - \pi A_j)x_j \\ &\textit{subject to } B_j x_j = b_j \\ &\qquad x_j \geq 0,\ j = 1,2 \end{aligned} \quad (44)$$

Obtaining solutions $x_j(\pi) = (\hat{x}_1, \hat{x}_2)$, $x_j(\pi) = (x^\wedge_1, x^\wedge_2)$ and optimal objective values $(\hat{z}_1, \hat{z}_2)$.

Step 2: Compute:

$$\theta_j = \hat{z}_1 - \hat{\pi}_j \ \theta_j = z^\wedge_j - \pi^\wedge_j,\ \textit{for } j = 1,2 \quad (45)$$

If for all j, $\theta_j \geq 0$, then the solution is optimal for the master program. Thus, the algorithm is terminated.

Otherwise ($\theta_j < 0$), form the new column as $\begin{pmatrix} A_j x_j(\pi) \\ 1 \end{pmatrix}$. Add this column to the basis and form a new basis and new prices using the rules of simplex method and return to Step1 (Tor, O.B., *et al.*, 2008).

Benders Decomposition:

Benders Decomposition is an attractive approach for solving some mixed variable programming problems. This section consists of precise Benders decomposition algorithm to solve mixed integer nonlinear programs and one example applying the algorithm is presented. Assume an original mixed integer (zero-one type) nonlinear programming problem of the form:

$$\begin{aligned} &\textit{Maximize}_{x,y,z}\ C_1 X + G(Y) + C_2 Z \\ &\textit{subject to } A_1 \leq b_1 \\ &\qquad A_2 X + H(Y) + A_3 Z \leq b_2 \\ &\qquad J(Y) + A_4 Z \leq b_3 \\ &\qquad X \in (0,1) \\ &\qquad Y, Z \geq 0 \end{aligned} \quad (46)$$

Where:

X Vector of $(0,1)$ integer variables;

C_1 Vector of objective function coefficients for the X variables;

A_1 Matrix of constraint coefficients for the X variables in constraints containing only X variables;

A_2 Matrix of constraint coefficients for the X variables in constraints linking X, Y and Z variables;

Y Vector of nonnegative real variables which contains nonlinear terms in either the objective function or the constraints; The set Y may be null, but either Y or Z must exist for the procedure to function.

$G(Y)$ Nonlinear (or possibly linear) function involving the Y variables in the overall objective function;

$H(Y)$ Set of nonlinear (or possibly linear) functions involving the Y variables those appear in constraints linking X, Y and Z variables. Either $H(Y)$ or A_3 must exist for this procedure to work.

$J(Y)$ Set of nonlinear (or possibly linear) functions involving the Y variables in constraints containing only Y and Z variables.

Z Vector of non-negative real variables that appear in the problem in a linear fashion; This vector may be null, but either Y or Z must exist for the program to work.

C_2 Vector of objective function coefficients for the Z variables;

A_3 Matrix of constraint coefficients for the Z variables in the constraints linking the X, Y and Z variables. Either $H(Y)$ or A_3 must exist for this procedure to work.

Journal of Applied Science and Agriculture, 9(1) January 2014, Pages: 11-25

A_4 Matrix of constraint coefficients for the Z variables in the constraints which contain only; Y and Z variables;

b_1 Vector of resource endowments for constraints containing only X variables;

b_2 Vector of resource endowments for constraints linking, X , Y and Z variables.

b_3 Vector of resource endowments for constraints containing only Y and Z variables;

This problem will henceforth be referred to as the original problem. The problem is characterized by (1) the presence of the integer variables (X) and (2) linearly separable functions between the integer and real variables. To simplify the problem, suppose the integer variables were fixed at some level $\underline{X}$. The original problem then reduces to:

$$\begin{aligned} &Maximize_{Y,Z}\ G(Y)+C_2Z+C_1X \\ &subject\ to\ H(Y)+A_3Z \le b_1 - A_2\underline{X} \\ &\qquad J(Y)+A_4Z_5 \le b_3 \\ &\qquad Y,Z \ge 0 \end{aligned} \tag{47}$$

This reduced problem (henceforth referred to an s the sub-problem) has a corresponding dual.

$$\begin{aligned} &Minimize_{Y,\gamma,\lambda}\ \gamma[b_2 - A_2\underline{X} - H(Y)] + \lambda[b_3 - J(Y)] - Y\begin{bmatrix} \nabla G(Y) \\ -\gamma\nabla H(Y) - \lambda\nabla J(Y) \end{bmatrix} \\ &subject\ to\ \gamma\nabla H(Y) + \lambda\nabla J(Y) \ge \nabla G(Y) \\ &\qquad \gamma A_3 + \lambda A_4 \ge C_2 \\ &\qquad Y,\gamma,\lambda \ge 0 \end{aligned} \tag{48}$$

Where:

γ is the dual variable associated with the constraints containing real and integer variables.

λ is the dual variable associated with the constraints containing only real variables.

$\nabla G(Y)$ is the gradient of $G(Y)$.

$\nabla H(Y)$ is the gradient of $H(Y)$.

$\nabla J(Y)$ is the gradient of $J(Y)$.

If an optimum solution exists for the primal of the sub-problem. An optimal set of Y , γ and λ values can be identified, all solution values being dependent on $\underline{X}$. The dual objective function will also match that identified for the primal problem. Thus, the dual sub-problem can be substituted for the primal sub-problem in the original problem as follows:

$$\begin{array}{c} Maximize_{c_1,x} \\ x,\gamma,\lambda,Y \\ Subject\ to \\ A_1X \le b_1 \\ X \in (0,1) \end{array} + \begin{bmatrix} Minimize_{\gamma,\lambda,Y}(C(Y) + \gamma[b_2 - A_2X - H(Y) + \lambda[b_3 - J(Y)] - \\ \gamma\nabla C(Y) - \gamma\nabla H(Y)\lambda\nabla J(Y) \\ Subject\ to\ \gamma\nabla H(Y) + \lambda\nabla J(Y) \ge \nabla C(Y) \\ \gamma A_3 + \lambda A_4 \ge C_2 \\ \gamma,\lambda,Y \ge 0 \end{bmatrix} \tag{49}$$

This problem is maximized over X , Y , γ and λ . Substituting Q for the dual sub-problem yields:

$$\begin{aligned} &Maximize\ C_1X + Q \\ &subject\ to\ A_1X \le b_1 \\ &\qquad Q \le W(X,Y,\gamma,\lambda) \\ &\qquad X \in (0,1)\ Y,\gamma,\lambda \in (feasible\,regin) \\ &\qquad Q\ is\,unbounded \end{aligned} \tag{50}$$

Where:

$$\begin{aligned} &W(X,Y,\gamma,\lambda) = \gamma[b_2 - A_2X] + \lambda b_3 + \\ &Minimum_{Y,\gamma,\lambda}[G(Y_i) - \lambda H(Y) - \lambda J(Y) - Y[\nabla G(Y) - \gamma\nabla H(Y) - \lambda\nabla J(Y)]] \end{aligned} \tag{51}$$

This formulation is known as the master problem. Restricting Y , γ , and λ to those values that satisfy all sub-problem constraints allows us to explicitly drop the sub-problem constraints from the model. Although in theory the formulation will produce an optimal solution, it is in fact an intractable model. Since an infinite number of $W(X,Y,\gamma,\lambda)$ solutions exist. The key to Benders Decomposition is to identify a relevant subset of $W(X,Y,\gamma,\lambda)$ values that will lead us to the optimum solution. To demonstrate how these relevant constraints might be identified, we define $V(X_i)$ as being equals to the value of objective function of sub-problem:

$$V(X_i) = Minimum_{Y_i,\gamma_i,\lambda_i} \left\{ \begin{array}{l} G(Y_i + \gamma_i[b_2 - A_2 X_i - H(Y_i)]) + \\ \lambda_i[b_3 - J(Y_i)] - Y_i \nabla G(Y_i) - \gamma_i \nabla H(Y_i) - \lambda_i \nabla J(H) \end{array} \right\} \tag{52}$$

Where Y_i, γ_i and λ_i are the optimal dual variables for X_i the equation can be rewritten as:

$$V(X_i) = \gamma_i b_2 - \gamma_i A_2 X_i + \lambda_i b_3 + Minimum_{Y_i,\gamma_i,\lambda_i} \{G(Y_i) - \gamma_i H(Y_i) - \lambda_i J(Y_i) - Y_i[\nabla G(Y_i) - \lambda_i \nabla H(Y_i) - \lambda_i \nabla J(H)]\} \tag{53}$$

Substituting this result to $W(X,Y_i,\gamma_i,\lambda_i)$ gives us:

$$\begin{array}{l} W(X,Y_i,\gamma_i,\lambda_i) = \gamma_i b_2 - \gamma_i A_2 X + \lambda_i b_3 + \\ V(X_i) - \gamma_i b_2 + \gamma_i A_2 X - \lambda_i b_3 \end{array} \tag{54}$$

$$W(X,Y_i,\gamma_i,\lambda_i) = \gamma_i A_2 + V(X_i) - \lambda_i A_2 X \tag{55}$$

The new master problem becomes:

$$\begin{array}{l} Maximize_{X,Y} \;\; C'Y - (r/2)Y'SY \\ Subject\ to\ \ A_1 X \le b_1 \\ \qquad \gamma_i A_2 X + Q \le V(X_i) + \gamma_i A_2 \underline{X}_i \\ \qquad X_1, X_2 \in (0,1)\ \ Q\ is\, unbounded \end{array} \tag{56}$$

The sub-problem is optimized for each $\underline{X}$ specified. With the resultant dual variables used to create a corresponding constraint ($\gamma_i A_A X + Q \le V(X_i) + \gamma_i A_2 \underline{X}_i$ known as a Benders cut) in the master problem. Through this problem new values of X are chosen. The sub-problem solution is optimal for the X set used in creating the problem, but is not optimal unless $\underline{X}$ is the optimal set of X values ($\underline{X}^*$) (Cong Liu, M. Shahidehpour, Yong Fu, Zuyi Li, 2009).

The Benders Algorithm:

Step 1: An initial vector for $\underline{X}$ is input. Initial values for the optimistic and conservative bounds are set, as is the convergence tolerance (ε) and iteration number ($M = 0$). The initial optimistic bound∞isfor a maximization problem, -∞ being the corresponding conservative bound.

Step 2: The initial sub-problem is solved after adjusting the sub-problem to reflect the $\underline{X}$ vector's impact on constraint right hand sides.

Step 3: The sub-problem objective function plus $C_1\underline{X}$ is compared to the previous conservative bound. If this sum represents an improvement over the previous conservative bound, it becomes the new conservative bound. The current $\underline{X}$ and associated sub-problem solutions are saved as the incumbent solution. The difference between conservative and optimistic bounds is then compared to the convergence tolerance (ε). If bound difference is less than or equal to ε , go to Step 7. If not, the iteration number is incremented by 1 and new Benders cut is formed, which is then added as another constraint in the master problem.

Step 4: The master problem is solved.

Step 5: The master problem objective function value becomes the new optimistic bound. The bonds are again checked for problem convergence. If the difference is less than or equal to the tolerance, go to Step 7. If not, the $b_2 - A_2\underline{X}_i$ right hand side in the sub-problem is computed based on the new master problem solution.

Step 6: The sub-problem is solved given the $\underline{X}$ solution from Step 4. One then returns to Step 3.

Step 7: The dual variables are calculated for the optimal solution. The incumbent solution from Step 3 is then printed and the procedure terminated (Khodaei, A., M. Shahidehpour, 2010; Khodaei, A., *et al*., 2010).

Decoupling Approach:

As sampling-based algorithms continue to improve along with computation power, it becomes increasingly feasible in practice to directly solve challenging planning problems under differential constraints. There are

Journal of Applied Science and Agriculture, 9(1) January 2014, Pages: 11-25

many situations, however, in which computing such solutions is still too costly due to expensive numerical integration, collision detection, and complicated obstacles in a high-dimensional state space. Decoupled approaches become appealing because they divide the big problem into modules that are each easier to solve. Ideally, we would like to obtain feedback plans on any state space in the presence of obstacles and differential constraints. This assumes that the state can be reliably measured during execution. A typical decoupled approach involves four modules:

1. Use a motion planning algorithm to find a collision-free path, $\tau : [0, 1] \rightarrow C_{free}$.
2. Transform τ into a new path τ' so that velocity constraints on C if there are any) are satisfied. This might, for example, ensure that the Dubin's car can actually follow the path. At the very least, some path-smoothing is needed in most circumstances.
3. Compute a timing function $\sigma: [0, t_F] \rightarrow [0, 1]$ for τ' so that $\tau' o \sigma$ is a time-parameterized path through C_{free} with the following requirement. The state trajectory $\tilde{x}$ must satisfy $\dot{x}= f(x(t),u(t))$ and $u(t) \in U(x(t))$ for all time, until u_T is applied at time t_F.
4. Design a feedback plan (or feedback control law), $\pi: X \rightarrow U$ that tracks $\tilde{x}$. The plan should attempt to minimize the error between the desired state and the measured state during execution (Fairman, F.W., 1988).

Given recent techniques and computation power, the significance of this approach may diminish somewhat; however, it remains an important way to decompose and solve problems. This decomposition is arbitrary. If every module can be solved, then it is sufficient for producing a solution; however, such decomposition is not necessary. At any step along the way, completeness may be lost because of poor choices in earlier modules. It is often difficult for modules to take into account problems that may arise later.

Case Study:

An example may aid in gaining a better understanding of the decomposition solution process. Assume a risk-averse farmer is interested in identifying how many acres of cotton and sorghum he should produce on his 200-acre farm. Complicating his decision is the government farm program in which he may want to participate. Assume the program requires a farmer interested in participating for a single crop to participate on all acres planted to that crop and to also participate on all acres planted to other government-supported crops on that farm. What combined crop mix-program participation strategy will maximize his utility in an E-V sense?

A mixed integer-nonlinear programming problem which addresses a form of this question is:

$$\begin{aligned}
&Maximize_{X,Y} \;\; C'Y - (r/2)Y'SY \\
&Subject\ to\ X_1 + X_2 = 1 \\
&\qquad -210X_1 + Y_{11} + Y_{12} \le 0 \\
&\qquad -210X_2 + Y_{21} + Y_{22} \le 0 \\
&\qquad Y_{11} + Y_{12} + Y_{21} + Y_{22} \le 200 \\
&\qquad 12Y_{11} + 6Y_{12} + 15Y_{21} + 8Y_{22} \le 2500 \\
&\qquad X_1, X_2 \in (0,1) \;\; Y_{11}, Y_{12}, Y_{21}, Y_{22} \ge 0
\end{aligned} \tag{57}$$

Where:

$$C = \begin{pmatrix} 60 \\ 20 \\ 30 \\ 40 \end{pmatrix} \; Y = \begin{pmatrix} Y_{11} \\ Y_{12} \\ Y_{21} \\ Y_{22} \end{pmatrix} \; S = \begin{pmatrix} 400 & 60 & 200 & 36 \\ 60 & 225 & 38 & 135 \\ 200 & 38 & 625 & 68 \\ 36 & 135 & 68 & 324 \end{pmatrix} \tag{58}$$

C is a vector of net returns (in dollars); S is a covariance matrix of net returns (in dollars); X_1 and X_2 represent participation and non-participation, respectively, in the government farm program; Y_{11} and Y_{12} represent cotton and sorghum produced under the program; Y_{21} and Y_{22} represent the same crops produced outside the program; and r represents the farmer's degree of risk aversion (0.00056). The iterative solution using the above algorithm is as follows:

Step 1: Set the initial parameters
Conservative Bound= -∞
Optimistic Bound= +∞
Iteration Number (M) = 0
Tolerance Level (ε) = 0
The initial integer solution is $X_1 = 1,\ X_2 = 0$.
(i.e., participation in the government program)
Step 2: The initial sub-problem becomes

$$\begin{aligned}
&\textit{Maximize } C'Y - (r/2)Y'SY \\
&\textit{Subject to } Y_{11} + Y_{12} \le 210 \\
&\quad Y_{21} + Y_{22} \le 0 \\
&\quad Y_{11} + Y_{12} + Y_{21} + Y_{22} \le 200 \\
&\quad 12Y_{11} + 6Y_{12} + 15Y_{21} + 8Y_{22} \le 2500 \\
&\quad Y_{11}, Y_{12}, Y_{21}, Y_{22} \ge 0
\end{aligned} \tag{59}$$

Solving this sub-problem yields the solution Objective = 7520.

$$\begin{aligned}
&Y_{11} = 200 \quad Y_{21} = 0 \quad \gamma_1 = 0 \quad \lambda_1 = 15.20 \\
&Y_{12} = 0 \quad Y_{22} = 0 \quad \gamma_2 = 20.77 \quad \lambda_2 = 0
\end{aligned} \tag{60}$$

Step3: Adding the integer component of the master problem objective function gives the resulting sub-problem solution

0 + 7520 = 7520

Conservative Bound = max (-∞, 7520) = 7520

Optimistic Bound= +∞

The latest sub-problem solution becomes the incumbent solution and is saved. Because of the difference between conservative and optimistic bounds is larger than ε. A Benders cut must be generated and the master problem solved. The iteration number is incremented to 1 and Benders cut formed.

$$\begin{aligned}
&Q \le 7520 - (0)(-210)(X_1) - (20.77)(-210)(X_2) \\
&+ (0)(-210)(X_1) + (20.77)(-210)(X_2)
\end{aligned} \tag{61}$$

Or

$$4362X_2 + Q \le 7520 \tag{62}$$

This cut indicates the predicted objective function of the dual problem, given these shadow prices, is 7520 + 4362 times the value of X_2. Note that if the original integer values were inserted into this cut (i.e., $X_1 = 1$, $X_2 = 0$), Q would equal the latest sub-problem objective function (7520). The cut suggests that the objective function could be increased by 4362 if X_2 were in the solution. The value 4362 is based on the shadow price for one unit of Y_{21} and (or) Y_{22} entering the solution times the number of total units of Y_{21} and Y_{22} allowed to enter the solution if $X_2 = 1\ (210)$. This value overstates the worth of X_2 in solution, because the third constraint in the sub-problem prohibits the sum of Y_{21} plus Y_{22} from exceeding 200. Further, the shadow price does not reflect the indirect effect of having X_2 in solution; i.e., X_2 in solution means X_1 must come out of solution.

Step 4: The Benders cut is added to the master problem, which is then solved.

$$\begin{aligned}
&\textit{Maximize}_{Q,X}\ Q \\
&\textit{Subject to } X_1 + X_2 = 1 \\
&-4362X_2 + Q \le 7520
\end{aligned} \tag{63}$$

The solution is: $Q = 11881,\ X_1 = 0,\ X_2 = 1$. The optimistic bound becomes 1188.

Step 5: This master problem solution suggests that non-participation in the program will result in an original problem objective function that will not exceed 11881. As we just indicated, however, the actual solution value will be below this amount. Because the difference between bounds is still greater than the tolerance level, a second sub-problem must be created and solved. The algorithm uses this new integer solution to create a new sub-problem and returns to Step 2.

Step 2: The new sub-problem is solved.

$$\begin{aligned}
&\textit{Maximize } C'Y - (r/2)Y'SY\\
&\textit{Subject to } Y_{11} + Y_{12} \le 0\\
&\quad Y_{21} + Y_{22} \le 210\\
&\quad Y_{11} + Y_{12} + Y_{21} + Y_{22} \le 200\\
&\quad 12Y_{11} + 6Y_{12} + 15Y_{21} + 8Y_{22} \le 2500\\
&\quad Y_{11}, Y_{12}, Y_{21}, Y_{22} \ge 0
\end{aligned} \tag{64}$$

The Solution is Objective= 4576, $\begin{matrix} Y_{11} = 0 & Y_{21} = 39.55 & \gamma_1 = 45.17 & \lambda_1 = 7.17 \\ Y_{12} = 0 & Y_{22} = 160.45 & \gamma_2 = 0.0 & \lambda_2 = 0.0 \end{matrix}$

Step 3: The objective function value from this latest sub-problem becomes the current challenger to the incumbent conservative bound (7520). Because of the challenger (4576) is less than the incumbent (7520). The incumbent sub-problem solution remains unchanged. Likewise, the difference between the conservative and optimistic bounds is also unchanged, necessitating creation of another Benders cut that is added to the master problem. The cut is:

$$\begin{aligned} &Q \le 4575 - (45.17)(-210)(X_1) - (0)(-210)(X_2) + \\ &(45.17)(-210)(X_1) + (0)(-210)(X_2) \end{aligned} \tag{65}$$

Or

$$-9486X_1 + Q \le 4575 \tag{66}$$

At this point all possible integer solution combinations have been considered and Benders cuts generated for them. The next master problem solution should, therefore, result in convergence. The master problem is:

$$\begin{aligned}
&\textit{Maximize}_{Q,X} \quad Q\\
&\textit{Subject to } X_1 + X_2 = 1\\
&\quad -4362X_2 + Q \le 7520\\
&\quad -9486X_1 + Q \le 4575
\end{aligned} \tag{67}$$

Step 4: The solution to the master problem is objective = 7520, $Q = 7520 \; X_1 = 1 \; X_2 = 0$.

Step 5: The master problem objective function becomes the new optimistic bound. Since the optimistic and conservative bound are the same, the iterative process is terminated. Thus, the solution is objective= 7520 and

$X_1 = 1 \;\; Y_{11} = 200 \;\;\; Y_{21} = 0$

$X_2 = 0 \;\; Y_{12} = 0 \qquad Y_{22} = 0$

This example illustrates how the algorithm constructs trial solutions using the master problem and then tests them on the sub-problem. Even if the sub-problem created by the master problem is not the best sub-problem solution, the information generated by the sub-problem is useful in helping the master problem learn about the overall problem. Note that a Benders cut was generated for each possible combination of X values. This is not uncommon for small problems but is rare in larger problems.

Conclusion:

In this paper, the decomposition methods have been presented. Decomposition methods are efficient methods to solve multi-variable optimization problems. We described two basic decomposition methods (primal and dual decomposition) and used them to construct Dantzing-wolfe and Benders decomposition techniques. Benders decomposition method is used in large scale optimization problems extensively and Dantzing-Wolf decomposition also Benders decomposition method is the most efficient technique to solve mixed integer programming (already, the most robust method to solve short-term hydrothermal coordination with ac modeling is Benders decomposition). Also decoupling method and its algorithm procedure is explained in the last section. Finally a case study is presented for understanding decomposition method process.

REFERENCES

Aganagic, M., S. Mokhtari, 1997. "Security constrained economic dispatch using nonlinear Dantzig-Wolfe decomposition", *IEEE Transactions on Power systems*, 12(1): 105-112.

Bo Lu, M. Shahidehpour, 2005. "Unit Commitment with flexible generating units", *IEEE Transactions on Power Systems*, 20(2): 1022-1034.

Cong Liu, M. Shahidehpour, Yong Fu, Zuyi Li, 2009. "Security-Constrained unit commitment with natural gas transmission constraints", *IEEE Transactions on Power Systems*, 24(3): 1523-1536.

Fairman, F.W., 1988. "Jordan Form Realization via Singular Value Decomposition", *IEEE Transaction on Circuits and System*s, 35(11).

Hoang Hai Hoc, 1982. "Topological optimization of networks: A nonlinear mixed integer model employing generalized Benders decomposition", *IEEE Transactions on Automatic Control*, 27(1): 164-169.

Jae Hyung Roh, M. Shahidehpour, Yong Fu, 2007. "Market-Based coordination of transmission and generation capacity planning", *IEEE Transactions on Power Systems*, 22(4): 1406-1419.

Jae Hyung Roh, M. Shahidehpour, Yong Fu, 2007. "Security-Constrained resource planning in electricity markets", *IEEE Transactions on Power Systems*, 22(2): 812-820.

Jianhui Wang, M. Shahidehpour, Zuyi Li, 2008. "Security-Constrained unit commitment with volatile wind power generation", *IEEE Transactions on Power Systems*, 23(3): 1319-1327.

Khodaei, A., M. Shahidehpour, 2010. "Transmission switching in security-constrained unit commitment", *IEEE Transactions on Power Systems*, 25(4): 1937-1945.

Khodaei, A., M. Shahidehpour, S. Kamalinia, 2010. "Transmission switching in expansion planning", *IEEE Transactions on Power Systems*, 25(3): 1722-1733.

Pennanen, H., A. Tolli, M. Latva-aho, 2011. "Decentralized coordinated downlink beamforming via Primal Decomposition", *IEEE Signal Processing Letters*, 18(11): 647-650.

Shahidehpour, M., Yong Fu, 2005. "Benders decomposition: applying benders decomposition to power systems", *IEEE Transactions on Power Systems*,3(2): 20-21.

Sifuentes, W.S., A. Vargas, 2007. "Hydrothermal scheduling using Benders decomposition: Accelerating techniques", *IEEE Transactions on Power Systems*, 22(3): 1351-1359.

Stephen Boyd, Lieven Vandenberghe, 2009. *Convex Optimization*, Cambridge University Press, New York, 7th Print.

Tor, O.B., A.N. Guven, M. Shahidehpour, 2008. "Congestion-Driven transmission planning considering the impact of generator expansion", *IEEE Transactions on Power Systems*, 23(2): 781-789.

Torresani, L., V. Kolmogorov, C. Rother, 2013. "A dual decomposition approach to feature correspondence", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 35(2): 259-271.

Yong Fu, M. Shahidehpour, 2007. "Fast SCUC for large-scale power systems", *IEEE Transactions on Power Systems*, 22(4): 2144-2151.

Yong Fu, M. Shahidehpour, Zuyi Li, 2005. "Security-Constrained Unit Commitment with AC Constraints", *IEEE Transactions on Power Systems*, 20(3): 1538-1550.

Yong Fu, M. Shahidehpour, Zuyi Li, 2006. "AC contingency dispatch based on security-constrained unit commitment", *IEEE Transactions on Power Systems*, 21(2): 897-908.

Zuyi Li, M. Shahidehpour, 2005. "Security-Constrained unit commitment for simultaneous clearing of energy and ancillary services markets", *IEEE Transactions on Power Systems*, 20(2): 1079-1088.