International Journal of Innovative Computing, Information and Control
Volume **8**, Number **10(A)**, October **2012**
 ISSN 1349-4198
pp. 6837–6848

STUDY ON STOCHASTIC PROGRAMMING METHODS BASED ON SYNTHESIS EFFECT

Chenxia Jin[1], Fachao Li[1], Yan Shi[2,3] and Lei Zhou[1]

[1]School of Economy and Management
Hebei University of Science and Technology
No. 70, Yuhua East Road, Shijiazhuang 050018, P. R. China
jinchenxia2005@126.com; lifachao@tsinghua.org.cn

[2]School of Industrial Engineering
Tokai University
9-1-1, Toroku, Kumamoto 862-8652, Japan
yshi@ktmail.tokai-u.jp

[3]School of Management
Dalian University of Technology
No. 2, Linggong Road, Ganjingzi District, Dalian 116024, P. R. China

Received July 2011; revised November 2011

Abstract. *Stochastic programming is a well-known optimization problem in resource allocation, optimization decision, etc. In this paper, we first analyze the essential features of stochastic programming and the deficiencies of the existing methods. To systematically process the objective function and constraints, we give the principles we should obey in seeking for the optimal decision, and further we give an axiomatic system for synthesis effect function. Based on synthesis effect function, we establish a general solution model (simply denoted as BSE-SGM) for stochastic programming problem, discuss the concavity of BSE-SGM, and analyze the effectiveness of BSE-SGM by an example. The results indicate that our method includes the existing stochastic programming methods, which can integrate the decision consciousness into the solution process effectively.*
Keywords: Stochastic programming, Synthesis effect, Uncertainty, Reliability, BSE-SGM

1. Introduction.

Randomness is a widespread uncertainty and unavoidable in many practice. Processing random information is very important in production management, artificial intelligence, complex systems optimization, etc. A random variable can be regarded as a family of data satisfying some laws, and there no order relation exists; therefore, the common programming methods are not suitable for stochastic programming problem. At present, there are three basic methods for the stochastic programming problem: 1) Expectation model [1]. Its basic idea is to describe a random variable with its mathematical expectation, and then convert a stochastic programming into a general one. 2) Chance-constrained model [2]. Its basic idea is to convert stochastic constraints and objective functions into ordinary ones respectively through some reliability principles. 3) Dependent-chance programming model [3]. Its basic idea is to regard objective functions and constraints as events under random environment, and then solve the stochastic programming problem by maximizing the probability of related events. These methods have good theoretical foundations, and also have been applied to many fields successfully. [4] studied measures program of oilfield by using expectation model, [5] established the expectation model for random transportation problems, and [6] considered the problem of producing and transporting a unique product directly from origin to a destination where demands are stochastic using expectation model, [7] used it to optimize allocation of harmonic filters on a distribution network, [8] studied disaster prevention flood emergency logistics planning problems using chance-constrained programming, [9] proposed a mixed chance-constrained programming model with randomness and fuzziness, [10] proposed possibilistic programming problems formed by a set of possibilistic constraint conditions with possibilistic ideal goals of decision variables, and the two possibility distributions were considered for reflecting the inherent uncertainty in the decision problems. However, these methods could not solve the stochastic programming under complicated environment effectively, and a couple of deficiencies still exist: 1) the expectations cannot effectively describe and represent random variables for extreme uncertainty, and it is difficult to ensure the reliability of expectation model; 2) when the stochastic features are complex (that is, the probability distributions of objective and constraints are often difficult to describe precisely), the computational complexity of the chance-constrained model and dependent-chance programming model are too high. In view of the above problems, many scholars have studied this problem from different aspects; for example, [11-15] constructed some solution methods through integrating random simulation and some intelligent algorithms (such as genetic algorithm, simulated annealing algorithm, and particle swarm optimization algorithm). However, random simulation must involve lots of tests, so these methods are only suitable for small-scale stochastic programming problem, and all have strong points. Until now, there still no systematic and effective stochastic programming methods exist.

In this paper, by analyzing the basic features of stochastic programming and the deficiencies of existing methods, we have the following contributions: a) we give the principles we should obey in processing the objective function and constraints, and then give an axiomatic system for synthesis effect function; b) based on the synthesis effect function, we establish a general solution model (simply denoted as BSE-SGM); c) we analyze the features of our model by an example, and the results indicate that our methods are effective.

In what follows, for the random variable ξ and event A on a probability space $(\Omega, ß, \Pr)$, let $E(\xi)$ and $D(\xi)$ be the mathematical expectation and variation of ξ, respectively, and $\Pr(A)$ be the probability of A.

2. **Overview of the Stochastic Programming Methods.** Stochastic programming involves in many fields such as production management, resource allocation etc. And the general mathematical form is

$$\begin{cases} \max f(x,\xi), \\ \text{s.t. } g_j(x,\xi) \le 0, \quad j = 1, 2, \cdots, m. \end{cases} \tag{1}$$

Here, $x = (x_1, x_2, \dots, x_n)$ is the decision vector, $\xi = (\xi_1, \xi_2, \dots, \xi_n)$ is the given random vector on $(\Omega, ß, \Pr)$, $f(x,\xi)$, $g_j(x,\xi)$ are random variable functions, $j = 1, 2, \cdots, m$.

As there no simple order exist between random variables, and $g_j(x,\xi) \le 0$ mostly could not be completely satisfied, model (1) is just a model in form and cannot be solved directly. It is necessary to convert (1) into ordinary one by some strategies.

1) **Expectation model**. We use the mathematical expectation to replace the random variable approximately, then model (1) can be converted into the following model (2):

$$\begin{cases} \max E(f(x,\xi)), \\ \text{s.t. } E(g_j(x,\xi)) \le 0, \quad j = 1, 2, \cdots, m. \end{cases} \tag{2}$$

Generally, model (2) is called the Expectation model [1]. When the variation is larger, the mathematical expectation could not describe the random variable effectively. So we could not get the optimal solution of the stochastic programming by model (2).

2) **Chance-constrained model**. The constraint of the stochastic programming cannot be often satisfied absolutely. If we use the reliability to deal with the constraints and objective functions, then model (1) can be converted into the following model (3):

$$\left\{\begin{array}{l}\max \overline{f}(x),\\ \text{s.t. } \Pr(f(x,\xi)\geq \overline{f}(x))\geq \alpha,\\ \Pr(g_j(x,\xi)\leq 0)\geq \alpha_j, \quad j=1,2,\cdots,m.\end{array}\right. \tag{3}$$

Generally, model (3) is called the chance-constrained model [2]. Here, α_j, $\alpha \in [0,1]$ are the reliabilities that the decision should satisfy. Compared with model (2), this model can control the quality of the decision beforehand, but it still cannot realize the solution if the distribution of $f(x,\xi)$ and $g_j(x,\xi)$ are complex or difficult to describe.

3) **Dependent-chance programming model**. Please see the following examples, a natural gas company has three gas stations (the maximum supplies are ξ_1, ξ_2, ξ_3, respectively, and they all exist uncertainty). The three stations supply gas for four living quarters (the demands are c_1, c_2, c_3, c_4, respectively). If the ith station supply x_{ij} unit gas for the jth living quarter, then $\Sigma_{i=1}^{3}x_{ij} = c_j$ is a random event, that is, $f_j(x) = \Pr\{\Sigma_{i=1}^{3}x_{ij} = c_j\}$ is a chance function, and it can be abstractly described as $f_j(x) = \Pr\{h_j(x,\xi)\leq 0\}$. Then we can get model (4):

$$\left\{\begin{array}{l}\max f(x) = \Pr\{\Sigma_{i=1}^{3}x_{ij} = c_j, j=1,2,3,4\},\\ \text{s.t. } \Sigma_{j=1}^{4}x_{ij}\leq \xi_i, \quad i=1,2,3,\\ \qquad x_{ij}\geq 0, \quad i=1,2,3, \quad j=1,2,3,4.\end{array}\right. \tag{4}$$

For the above problems, [3] proposed dependent-chance programming, that is

$$\left\{\begin{array}{l}\max \Pr(h_k(x,\xi))\leq 0, \quad k=1,2,\cdots,q,\\ \text{s.t. } g_j(x,\xi)\leq 0, \quad j=1,2,\cdots,p.\end{array}\right. \tag{5}$$

Here, $x=(x_1,x_2,\cdots,x_n)\in R^n$ is decision variable, $\xi=(\xi_1,\xi_2,\cdots,\xi_n)$ is a given random variable vector in $(\Omega, ß, \Pr)$, $h_k(x,\xi)$ is a chance function, $g_j(x,\xi)$ is a chance constraint.

Stochastic programming is an uncertain decision problem, the results cannot generally satisfy the related constraints absolutely. Therefore, it is more suitable synthetically considering the constraint satisfaction and the size of objective. To establish a general solution model under this idea, we can synthesize the objective and constraint satisfaction through some strategies (called the synthesis effect function), then discuss the programming based on the synthesis effect value. Following will give the axiomatic system for multi-attribute random synthesis effect functions.

3. The Axiomatic System for Synthesis Effect Function.

In the following, we mainly discuss the synthesis strategy of objective and the constraints satisfaction of the maximization stochastic programming problem with single objective (real function) and multi-constraints. According to the essential features of the optimal decision, we should obey the following principles in seeking for the decision scheme of stochastic programming.

Principle 1 If the satisfaction degree of constraints is same, the greater the objective is, the better the effect is;

Principle 2 If the objective is same, the greater the satisfaction degree of constraints is, the better the effect is;

Principle 3 If the constraints are absolutely satisfied, the decision only depends on the objective;

Principle 4 If some constraints are absolutely dissatisfied, we cannot make a decision.

If we abstractly regard the synthesis strategy of objective and satisfaction degree of constraints as a function $S(u,v) = S(u,v_1,v_2,\cdots,v_m)$ (here, u is the objective with the conversion interval Θ; v_i is the satisfaction degree of the ith constraint with the conversion interval $[0,1]$, that is, $S(u,v)$ is a map on $\Theta \times [0,1]^m \to (-\infty,+\infty)$), then the above principles can be equivalent to the following four conditions:

Condition 1 For any given $u \in \Theta$, $S(u,v)$ is monotone non-decreasing in each v_i;

Condition 2 For any given $v = (v_1, v_2, \cdots, v_m) \in [0,1]^m$, $S(u,v)$ is monotone non-decreasing in u;

Condition 3 $S(u,1,1,\cdots,1)$ is strictly monotone increasing;

Condition 4 When $\Pi_{j=1}^m v_j = 0$, $S(u_1, v_1, v_2, \cdots, v_m) = S(u_2, v_1, v_2, \cdots, v_m)$ for any $u_1, u_2 \in \Theta$.

Obviously, for uncertain decision problem, Principles 1-3 must be obeyed, while Principle 4 can be loosed. For convenience, we call $S(u,v)$ satisfying Conditions 1-3 synthesis effect function on Θ, and $S(u,v)$ satisfying Conditions 1-4 uniform synthesis effect function on Θ.

Remark 3.1. *For multi-objective decision, the above four principles must be also obeyed, so we can similarly establish the axiomatic system for its synthesis effect function.*

According to the above definition, we have the following conclusions:

1) For any given $a < b \in (-\infty,+\infty)$, $S(u, v_1, v_2, \cdots, v_m) = (u-a)(b-a)^{-1} \wedge v_1 \wedge v_2 \wedge \cdots \wedge v_m$ is a uniform synthesis effect function on $[a,b]$. Here, $\wedge$ is min operation of real numbers.

2) For any given $k \in (0,+\infty)$, $c \in [0,+\infty)$, $\alpha_j \in (0,+\infty)$, $S(u, v_1, v_2, \cdots, v_m) = k(u+c)\Pi_{j=1}^m v_j^{\alpha_j}$ is a uniform synthesis effect function on $[0,+\infty)$.

3) For any given $\alpha \in (0,\infty)$, $\beta_j \in [1,\infty)$, $S_1(u, v_1, v_2, \cdots, v_m) = u^\alpha \Pi_{j=1}^m v_j^{\beta_j}$ is a uniform synthesis effect function on $[0,+\infty)$; $S_2(u, v_1, v_2, \cdots, v_m) = \exp(\alpha u)\Pi_{j=1}^m v_j^{\beta_j}$ is a uniform synthesis effect function on $(-\infty,+\infty)$.

4) For any given $\alpha \in (0,\infty)$, $\lambda_j \in (0,1]$, $S_1(u, v_1, v_2, \cdots, v_m) = u\Pi_{j=1}^m \delta(v_j - \lambda_j)$ is a uniform synthesis effect function on $[0,+\infty)$; $S_2(u, v_1, v_2, \cdots, v_m) = \exp(\alpha u) \times \Pi_{j=1}^m \delta(v_j - \lambda_j)$ is a uniform synthesis effect function on $(-\infty,+\infty)$; $S_3(u, v_1, v_2, \cdots, v_m) = \alpha u + \Sigma_{j=1}^m \eta(v_j - \lambda_j)$ is a uniform synthesis effect function on $(-\infty,+\infty)$. Here, $\delta(t) = 0$ for $t < 0$, and $\delta(t) = 1$ for $t \ge 0$; $\eta(t) = -\infty$ for $t < 0$, and $\eta(t) = 1$ for $t \ge 0$.

5) For any given $a, b, \alpha_j \in (0,+\infty)$, $j = 1, 2, \cdots, m$, $S(u, v_1, v_2, \cdots, v_m) = bu + a\Sigma_{j=1}^m \alpha_j v_j$ is a synthesis effect function on $(-\infty,+\infty)$, but is not uniform.

4. **Stochastic Programming Model Based on Random Synthesis Effect Function (BSE-SGM).**

4.1. **Description of BSE-SGM.**

Using the above synthesis strategy, for stochastic programming problem (1), if we regard u and v_j in $S(u, v_1, v_2, \cdots, v_m)$ as the concentrated quantification value $C(f(x,\xi))$ of $f(x,\xi)$ and the satisfaction degree $\beta_j(x) = \mathrm{Sat}(g_j(x,\xi) \le 0)$ of the ith constraint for the scheme x, respectively, then $S(C(f(x,\xi))$, $\beta_1, \beta_2, \cdots, \beta_m)$, considering both objective and constraint, is a quantitative descriptive model measuring the quality of the solution, so model (1) can be converted into the following model (6):

$$\begin{cases} \max S(C(f(x,\xi)), \beta_1(x), \beta_2(x), \cdots, \beta_m(x)), \\ \text{s.t. } x \in X. \end{cases} \tag{6}$$

Here, $(\inf C(f(x,\xi)), \sup C(f(x,\xi))) \subset \Theta$, $S(u, v_1, v_2, \cdots, v_m)$ is a (uniform) synthesis effect function on Θ. For convenience, we call (6) stochastic programming model based on synthesis effect, simply denoted as BSE-SGM.

4.2. **Features of BSE-SGM.**

Theorem 4.1. *If model (1) is a crisp programming,* $(\inf f(x,\xi), \sup f(x,\xi)) \subset \Theta$*,* $S(u, v_1, v_2, \cdots, v_m)$ *is a uniform random synthesis effect function on* Θ*. When the number of elements in* $\cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$ *is more than 1, and* $f(x,\xi)$ *is not constant function on* $\cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$*, then model (6) and model (1) have the same optimal solution.*

Proof: As the constraint of a crisp programming only has two states: satisfied or dissatisfied, $\beta_j(x)$ only has two values: 0 or 1, that is, $\beta_j(x) = \mathrm{Sat}(g_j(x,\xi) \le 0) = 1$ for $g_j(x,\xi) \le 0$, and $\beta_j(x) = \mathrm{Sat}(g_j(x,\xi) \le 0) = 0$ for $g_j(x,\xi) > 0$.

1) If x^* is the optimal solution of model (1), then $\beta_j(x) = \mathrm{Sat}(g_j(x,\xi) \le 0) = 1$, $j = 1, 2, \cdots, m$. In the following, we will prove $S(C(f(x,\xi)), \beta_1(x), \beta_2(x), \cdots, \beta_m(x)) \le S(C(f(x^*,\xi)), \beta_1(x^*), \beta_2(x^*), \cdots, \beta_m(x^*))$ through two cases.

a) For any $x \in \cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$, by $f(x,\xi) \le f(x^*,\xi)$, $\beta_j(x) = \mathrm{Sat}(g_j(x,\xi) \le 0) \equiv 1$, $D(f(x,\xi)) = 0$, and the monotonicity increasing of $C(f(x,\xi))$ and $S(u, 1, 1, \cdots, 1)$, we have that $S(C(f(x,\xi)), \beta_1(x), \beta_2(x), \cdots, \beta_m(x)) \le S(C(f(x,\xi)), 1, 1, \cdots, 1) \le S(C(f(x^*, \xi)), 1, 1, \cdots, 1) = S(C(f(x^*,\xi)), \beta_1(x^*), \beta_2(x^*), \cdots, \beta_m(x^*))$.

b) For any $x \in X - \cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$, at least there exists a $j \in \{1, 2, \cdots, m\}$ such that $\beta_j(x) = \mathrm{Sat}(g_j(x,\xi) \le 0) = 0$, so $\Pi_{j=1}^m \beta_j(x) = 0$, combining with the properties of $S(u, v_1, v_2, \ldots, v_m)$, we know that $S(C(f(x,\xi)), \beta_1(x), \beta_2(x), \ldots, \beta_m(x)) = S(C(f(x,\xi)), 0, 0, \cdots, 0) \le S(C(f(x^*,\xi)), 1, 1, \cdots, 1) = S(C(f(x^*,\xi)), \beta_1(x^*), \beta_2(x^*), \cdots, \beta_m(x^*))$.

2) If x^* is the optimal solution of model (6), then $S(C(f(x,\xi)), \beta_1(x), \beta_2(x), \cdots \beta_m(x)) \le S(C(f(x^*,\xi)), \beta_1(x^*), \beta_2(x^*), \cdots, \beta_m(x^*))$ for any $x \in X$. Since $S(u, 1, 1, \ldots, 1)$ is strictly monotone increasing, we only prove that $x^* \in \cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$, then we can obtain that x^* is the optimal solution of model (1).

Actually, if $x^* \notin \cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$, then there must exist j such that $\beta_j(x^*) = \mathrm{Sat}(g_j(x^*,\xi) \le 0) = 0$, by this, we have that $S(C(f(x^*,\xi)), \beta_1(x^*), \beta_2(x^*), \cdots, \beta_m(x^*)) \le S(C(f(x,\xi)), \beta_1(x), \beta_2(x), \cdots, \beta_m(x))$ for any $x \in X$, that is $S(C(f(x^*,\xi)), \beta_1(x^*), \beta_2(x^*), \cdots, \beta_m(x^*)) = S(C(f(x,\xi)), \beta_1(x), \beta_2(x), \cdots, \beta_m(x))$ for any $x \in X$, this implies that $S(f(x), \beta_1(x), \beta_2(x), \cdots, \beta_m(x))$ is a constant function or $-\infty$ on X. Particularly, $S(C(f(x,\xi)), \beta_1(x), \beta_2(x), \cdots, \beta_m(x))$ is a constant function or $-\infty$ on $\cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$. By this and the number of elements in $\cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$ is more than 1 and $S(u, 1, 1, \cdots, 1)$ is strictly monotone increasing, we know that $S(f(x,\xi), 1, 1, \cdots, 1)$ is a constant function on $\cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$, this contradicts to the conditions.

In fact, most crisp programming problems satisfy that the number of elements in $\cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$ is more than 1 and $f(x,\xi)$ is not constant function on $\cap_{j=1}^m \{w | g_j(w,\xi) \le 0\}$. So model (6) can be considered as an extension of ordinary programming problem. However, if $f(x,\xi)$ or $g_j(x,\xi)$ exist randomness, the optimal solution determined by different (uniform) synthesis effect functions is generally not same, sometimes even this difference is great (see Section 5), and it can be popularly interpreted that this difference is caused by different decision consciousness.

The above analysis indicate that, model (6) is the extension of the above three stochastic programming problems discussed in Section 2. And, if we select different synthesis effect function, model (6) can also be converted into the above three stochastic programming problems, see the following Remarks 4.1-4.3.

Remark 4.1. *If we use* $E(f(x,\xi))$ *to concentralizedly describe the size of* $f(x,\xi)$*,* $\beta_j(x) = \delta(E(-g_j(x,\xi)))$ *the satisfaction degree of* $g_j(x,\xi) \le 0$*, then model (6) is the expectation model (2) for* $S(u,v) = \exp(u)\eta(\Pi_{j=1}^m \delta(v_j - \alpha_j))$*. Here,* $\delta(t) = 0$ *for* $t < 0$*, and* $\delta(t) = 1$ *for* $t \ge 0$*; and* $\eta(0) = -\infty$*,* $\eta(1) = 1$*.*

Remark 4.2. *If we use $\overline{f}(x)$ to concentralizedly describe the size of $f(x,\xi)$, $\beta_j(x) = \Pr(g_j(x,\xi) \le 0)$ the satisfaction degree of $g_j(x,\xi) \le 0$, then model (6) is the chance-constrained model (3) for $S(u,v) = \exp(u)\eta(\Pi_{j=1}^{m}\delta(v_j - a_j))$. Here, $\overline{f}(x) = \max\{\tau(x)|\Pr(f(x,\xi) \ge \tau(x)) \ge \alpha\}$, and $\delta(t) = 0$ for $t < 0$, $\delta(t) = 1$ for $t \ge 0$; $\eta(0) = -\infty$, $\eta(1) = 1$.*

Remark 4.3. *If we use $f_i(x) = \Pr(h_i(x,\xi) \le 0)$ to represent the satisfaction degree of $h_i(x,\xi) \le 0$, $i = 1,2,\cdots,q$, and $f_i(x) = \Pr(g_i(x,\xi) \le 0)$ the satisfaction degree of $g_i(x,\xi) \le 0$, $i = q+1, q+2, \cdots, q+p$, then model (6) is the dependent-chance programming model (5) for $S(u,v) = \Pi_{i=1}^{q} v_i \cdot H(\Pi_{i=q+1}^{q+p} v_i)$. Here, $H(0) = 0$, and $H(t) = 1$ for $t > 0$.*

These discussions indicate that our model is effective not only in theory, but also in practice. It can embody the non-variance of stochastic programming, it has better structural characteristic and strong interpretation, therefore model (6) provides a theoretical platform for solving stochastic programming problem. Remark 4.4 will give us an operable method for model (6).

Remark 4.4. *If we use $(E(f(x,\xi)), D(f(x,\xi)))$ to describe the compound quantification of $f(x,\xi)$ from the size characteristic, and $T(E(f(x,\xi)), D(f(x,\xi)))$ to represent $C(f(x,\xi))$ (Here, $T(E(f(x,\xi)), D(f(x,\xi)))$ is some synthesis value of $E(f(x,\xi))$ and $D(f(x,\xi))$, and $T(x,y)$ satisfies: 1) $T(s,t)$ is monotone non-decreasing on s; 2) $T(s,t)$ is monotone non-increasing on t, 3) $T(s,0)$ is increasing on s), and use $\beta_j(x) = \Pr(g_j(x,\xi) \le 0)$ to represent the satisfaction degree of $g_j(x,\xi) \le 0$, then the model (6) can be*

$$\begin{cases} \max S(T(E(f(x,\xi)), D(f(x,\xi))), \beta_1(x), \beta_2(x), \cdots, \beta_m(x)), \\ \text{s.t. } x \in X. \end{cases} \tag{7}$$

Remark 4.5. *In practice, most constraints in stochastic programming cannot be absolutely satisfied, sometimes we pay more attention to the importance degree to each constraint, please see the example given in introducing Dependent-chance programming in Section 2. Through pre-setting a threshold value α that is the satisfaction degree to each constraint, the existing stochastic programming methods could to some entent reflect the importance degree to each constraint when making decision, but this limit is too absolute and also non-continuous, while our model can not only embody the difference in importance for each constraint during the process of decision, also reflect the continuity of decision consciousness through the selection of synthesis effect function (including the form and parameters), please see the example in Section 5.*

4.3. **Convexity of BSE-SGM.** In the following, we will discuss the convexity of BSE-SGM. First, we will introduce the concept of stochastic convex (concave).

Definition 4.1. [16] *Let $X \subset R^n$ be a convex set, ξ be a given random variable on probability space $(\Omega, \text{ß}, \Pr)$, $f(x,\xi)$ be a random variable function on X.*

1) If $D(f(\lambda x_1 + (1-\lambda)x_2), \xi) \le \lambda D(f(x_1,\xi)) + (1-\lambda)D(f(x_2,\xi))$ for any given $x_1, x_2 \in X$ and $\lambda \in [0,1]$, then $f(x,\xi)$ is stochastic convex with respect to x on X; 2) If $D(f(\lambda x_1 + (1-\lambda)x_2), \xi) \ge \lambda D(f(x_1,\xi)) + (1-\lambda)D(f(x_2,\xi))$ for any given $x_1, x_2 \in X$ and $\lambda \in [0,1]$, then $f(x,\xi)$ is stochastic concave with respect to x on X.

Theorem 4.2. *Let $X \subset R^n$ be a convex set, ξ be a given random variable on probability space $(\Omega, \text{ß}, \Pr)$, $f(x,\xi)$ be a random variable function on X.*

1) If $f(x,\xi) = h(x)\varphi(\xi)$, and $h(x)$ is a non-negative convex function on X, then $f(x,\xi)$ is stochastic convex with respect to x on X. 2) If $f(x,\xi) = h(x)\varphi(\xi)$, and $h^2(x)$ is a concave function on X, then $f(x,\xi)$ is stochastic concave with respect to x on X. 3) If $f(x,\xi) = h(x) + \varphi(\xi)$, then $f(x,\xi)$ is not only stochastic convex, but also stochastic concave with respect to x on X. 4) For any real number a, $af(x,\xi)$ has the same convexity or concavity with $f(x,\xi)$.

It can be proved by the definition of convex and concave and the properties of variance.

Theorem 4.3. [16] *If random variable function $g(x, \xi)$ is joint convex with respect to (x, ξ), and probability measure* Pr *is pseudo-concave, then for any given confidence level $\alpha \in [0, 1]$, $B = \{x | \Pr\{g(x, \xi) \leq 0\} \geq \alpha\}$ is a convex set.*

Theorem 4.4. *Let $X \subset R^n$ be a convex set, ξ be a given random variable on probability space $(\Omega, ß, \Pr)$. If: 1) For any state of ξ, $f(x, \xi)$ is convex with respect to x on X; 2) $f(x, \xi)$ is stochastic concave with respect to x on X; 3) $T(s, t)$ is joint convex with respect to (s, t); 4) $g_j(x, \xi)$ is joint convex with respect to (x, ξ), $j = 1, 2, \cdots, m$; 5) Probability measure* Pr *is pseudo-concave, then (8) is a convex programming.*

Proof: Denote $B_j = \{x | \Pr\{g_i(x, \xi) \leq 0\} \geq \alpha_j\}$, $j = 1, 2, \cdots, m$. From condition 4), 5) and Theorem 4.3, we know B_j are all convex set, so $B = \cap_{j=1}^{m}(B_j \cap X)$ is convex set. In the following, we will prove $T(E(f(x, \xi)), D(f(x, \xi)))$ is convex with respect to x on B.

For any given $x_1, x_2 \in B$ and $\lambda \in [0, 1]$, by 1), we know $f(\lambda x_1 + (1 - \lambda)x_2, \xi) \leq \lambda f(x_1, \xi) + (1 - \lambda)f(x_2, \xi)$ and $E(f(\lambda x_1 + (1 - \lambda)x_2, \xi)) \leq \lambda E(f(x_1, \xi)) + (1 - \lambda)E(f(x_2, \xi))$, that implies that $E(f(x, \xi))$ is convex. Using the properties of $T(s, t)$ and condition 2), 3), we know that $T(E(f(\lambda x_1 + (1 - \lambda)x_2, \xi)), D(f(\lambda x_1 + (1 - \lambda)x_2, \xi))) \leq T(E(f(\lambda x_1 + (1 - \lambda)x_2, \xi)), \lambda D(f(x_1, \xi)) + (1 - \lambda)D(f(x_2, \xi))) \leq T(\lambda E(f(x_1, \xi)) + (1 - \lambda)E(f(x_2, \xi)), \lambda D(f(x_1, \xi)) + (1 - \lambda)D(f(x_2, \xi))) \leq \lambda T(E(f(x_1, \xi)), D(f(x_1, \xi))) + (1 - \lambda)T(E(f(x_2, \xi)), D(f(x_2, \xi)))$. So $T(E(f(x, \xi)), D(f(x, \xi)))$ is convex with respect to x on B.

Remark 4.6. *Though Theorem 4.2 gives us some methods to judge the stochastic convex or stochastic concave of $f(x, \xi)$, a lot of cases in real life do not satisfy the Theorem 4.4, for instance, $f_1(x, \xi) = x^{\alpha}\xi$ are not stochastic concave for $\alpha \geq 0.5$.*

The above theorems give some good properties of BSE-SGM from the theoretical level. The fact reminds us using the analytical methods such as Newton method and simplex method. for the solution to convex programming model (6). However, all these conclusions need too strong conditions, BSE-SGM is still commonly not convex programming and we cannot solve it through traditional methods. Therefore we can solve it by using some intelligent algorithms such as genetic algorithm, and ant colony optimization algorithm.

5. **Example Analysis.**

In this Section, we will further analyze the features of stochastic programming model by a measure programming problem of oilfield.

Example 5.1. *In the mid to late part of the oilfield mining process, the appropriate measures (e.g., fracturing, acidizing, fill holes, plugging, transfer pumping, sand and heat mining) to maintain wells are important to extend the field production life, lower oil and gas costs and enhance the production rate. Due to that the exploitation of oil fields are often subject to many random factors, and therefore, the oil field measure programming can be described by the following stochastic programming problem:*

$$\begin{cases} \max z = \sum_{i=1}^{n} q_i x_i \\ \text{s.t. } \sum_{i=1}^{n} c_i x_i \leq T_C, \\ \qquad M^- \leq \sum_{i=1}^{n} x_i \leq M^+, \\ \qquad M_i^- \leq x_i \leq M_i^+, \quad i = 1, 2, \cdots, n, \\ \qquad x_i \geq 0, \text{ and is an integer}, \quad i = 1, 2, \cdots, n. \end{cases} \tag{8}$$

Here, n is the numbers of the measures; x_i is the ith measure's workload; T_C are the total cost of the measure programming, M^- and M^+ are the minimum of the total workload and the maximum of the total workload respectively; M_i^- and M_i^+ are the minimum and maximum of the ith measure's workload respectively; q_i, c_i are the unit production increment and the unit production cost for the ith measure. And q_i is a random variable with the uniform distribution $U(q^-, q^+)$; c_i is a random variable with the uniform distribution $U(c^-, c^+)$.

In model (8), we consider the following programming problem of the oilfield (denote SOF for short) like that: 1) maintenance measures include: fracturing, acidification, reperforating, water shutoff, transfer pumping, sand control, thermal mining, overhaul and others; 2) the total cost of the measures $T_C = 45000$ yuan; 3) the total workload of the measures is not more than 5160 times for individual well; 4) the basic data of the measures are presented in Table 1 and Table 2. So the measure programming is

$$\begin{cases} \max z = \sum_{i=1}^{9} q_i x_i \\ \text{s.t. } \sum_{i=1}^{9} c_i x_i \leq 45000, \\ \quad \sum_{i=1}^{9} x_i \leq 5160, \\ \quad M_i^- \leq x_i \leq M_i^+, \quad i = 1, 2, \cdots, 9, \\ \quad x_i \geq 0, \text{ and is an integer}, \quad i = 1, 2, \cdots, 9. \end{cases} \tag{9}$$

For uniform distribution $U(a, b)$, using the expect value $(a + b)/2$ and variance $(b - a)^2/12$, we can easily get Table 3. In order to analyze the features of BSE-SGM, we only solve the problem (9) by using the expectation model (2) and the BSE-SGM as follows.

TABLE 1. The parameters of $U(q_i^-, q_i^+)$ and $U(c_i^-, c_i^+)$

measures	increment production for individual well (ton)		cost for individual well (ten thousand yuan)	
	q^-	q^+	c^-	c^+
fractueing	1591	1619	29.93	36.85
acidificatio	2857	3000	53.74	68.2
reperforating	554	586	10.42	13.34
water shutoff	200	215	3.76	4.89
transfer pumping	759	867	15.7	19.73
sand control	77	80	1.45	1.82
thermal mining	430	436	8.2	9.79
overhaul	283	320	6.02	6.44
others	57	86	0.88	1.96

TABLE 2. The minimum and maximum of the every measure's workload

measures	minimum M_i^-	maximum M_i^+
fractueing	180	210
acidification	50	70
reperforating	120	140
water shutoff	1000	1200
transfer pumping	120	150
sand control	350	500
thermal mining	1980	2140
overhaul	460	600
others	0	350

TABLE 3. The mathematical expectation and variance of q_i and c_i

measures	$E(q_i)$	$E(q_i)$	$D(q_i)$	$D(c_i)$
fractueing	1605	33.39	65.333333	3.9905333
acidification	2928.5	60.97	1704.0833	17.4243
reperforating	570	11.88	85.333333	0.7105333
water shutoff	207.5	4.325	18.75	0.1064083
transfer pumping	813	17.715	972	1.3534083
sand control	78.5	1.635	0.75	0.0114083
thermal mining	433	8.995	3	0.210675
overhaul	301.5	6.23	114.08333	0.0147
others	71.5	1.42	70.083333	0.0972

(**I**) By model (2), (9) can be transformed into the general programming problem (10):

$$\begin{cases} \max z^* = 1605x_1 + 2928.5x_2 + 570x_3 + 207.5x_4 + 813x_5 + 78.5x_6 \\ \qquad\qquad +433x_7 + 301.5x_8 + 71.5x_9 \\ \text{s.t. } 33.9x_1 + 60.97x_2 + 11.88x_3 + 4.325x_4 + 17.715x_5 + 1.635x_6 + 8.995x_7 \\ \qquad +6.23x_8 + 1.42x_9 \le 45000, \\ \qquad \sum_{i=1}^{9} x_i \le 5160, \\ \qquad M_i^- \le x_i \le M_i^+, \quad i = 1, 2, \cdots, 9, \\ \qquad x_i \ge 0, \text{ and is an integer}, \quad i = 1, 2, \cdots, 9. \end{cases} \tag{10}$$

(**II**) By using BSE-SGM: its parameters settings are: 1) use $C(z) = E(z)[1 + (\max\sqrt{D(z)} - \sqrt{D(z)}) \div (\max\sqrt{D(z)} - \min\sqrt{D(z)})]^a$ to describe the concentrated quantification value of z (by Table 3, we know $\max\sqrt{D(z)} = 11196.21581$, $\min\sqrt{D(z)} = 8737.60365$); 2) use $S(u,v) = uv^b$ to describe the stochastic synthesis effect function, then problem (9) can be transformed into the following general programming problem (11):

$$\begin{cases} \max z'' = (1605x_1 + 2928.5x_2 + 570x_3 + 207.5x_4 + 813x_5 + 78.5x_6 + 433x_7 \\ \qquad\qquad + 301.5x_8 + 71.5x_9) * \beta^b[(11196.21581 - (65.333333x_1^2 + 1704.0833x_2^2 \\ \qquad\qquad + 85.333333x_3^2 + 18.75x_4^2 + 85.333333x_3^2 + 18.75x_4^2 + 972x_5^2 + 0.75x_6^2 \\ \qquad\qquad + 3x_7^2 + 114.08333x_8^2 + 70.083333x_9^2)^{0.5})/2458.61276]^a \\ \text{s.t. } \beta = \Pr\{c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + c_5x_5 + c_6x_6 + c_7x_7 + c_8x_8 + c_9x_9 \\ \qquad\qquad \le 45000\}, \\ \sum_{i=1}^{9} x_i \le 5160, \quad M_i^- \le x_i \le M_i^+, \quad i = 1, 2, \cdots, 9, \\ x_i \ge 0, \text{ and is an integer}, \quad i = 1, 2, \cdots, 9. \end{cases} \tag{11}$$

Obviously, a and b are the parameters describing the uncertainty consciousness. For any given $\beta \in (0,1)$, we have $\beta^b \to 0 (b \to +\infty)$, $\beta^b \to 1 (b \to +0)$, so b should not be too big, nor too small, and its best value generally be limited to between 0-3. The greater (smaller) b is, the greater (smaller) the satisfaction for constraints is. For the two solutions x and y for programming model (9). If the compound quantification values of the objective z are respectively $C(z(x)) = 3500000$, $C(z(y)) = 3300000$, and the levels of satisfaction for constraint $\sum_{i=1}^{9} c_i x_i \le 45000$ are respectively $\beta(x) = 0.8$, $\beta(y) = 0.84$, then we have: 1) when $b = 1$, the synthesis effect value of x and y are $z^*(x) = 3500000 \times 0.8 = 2800000$, $z^*(y) = 3300000 \times 0.84 = 2772000$ respectively, i.e., solution x is better than solution y; 2) when $b = 2$, the synthesis effect value of x and y are respectively $z^*(x) = 3500000 \times 0.8^2 = 2240000$, $z^*(y) = 3300000 \times 0.84^2 = 2328480$, i.e., y is better than x. Similarly, for $x > 0$, we know that $(1 + x)^a \to +\infty(a \to +\infty)$. Therefore, a should not be too big and its best value generally be limited to between 0-2. The smaller a is, the smaller the required level of uncertainty for objective is (for example, when $a = 0$, the quality of solution is irrelevant to the variance of objective); the greater a is, the greater the required level of uncertainty for objective is.

From the above analysis, we should synthetically consider both the objective and constraints, this is also required by the essence of decision. Otherwise, we may lose better solution. And our model can realize the idea, while the existing stochastic programming methods cannot.

Obviously, (10) and (11) are nonlinear programming problems and they cannot be solved easily by analytical methods. We can give the solution strategy combining with genetic algorithm, and its parameters setting are: binary code mode; mutation probability is 0.001; crossover probability is 1; population size is 80; evolutionary generations is 100. Then the results of the problem under different methods are shown in Table 4 (Here, S.E.V. denotes the Synthesis Effect Value).

Table 4. The results of the problem (10) under different methods

Solving Model		Optimal Solution	S.E.V.	Expectation of objective	Variation of objective	Constraint satisfaction
Expectation Model		(209, 70, 137, 1192, 147, 491, 2133, 585, 196)	———	2137905.0	10771.025	0.570
BSE-SGM	a=2, b=2	(206, 51, 124, 1008, 120, 491, 2013, 464, 30)	7258181.43	1909597.0	8861.54	0.989
	a=1, b=2	(206, 51, 124, 1008, 120, 491, 2013, 464, 30)	7258181.43	1909597.0	8861.54	0.989
	a=0.5, b=2	(209, 69, 134, 1008, 124, 467, 2121, 460, 28)	2723354.26	2019608.0	9184.24	0.981
	a=0.1, b=2	(210, 70, 132, 1006, 12, 492, 2137, 492, 75)	2163012.99	2042046.5	9371.04	0.979
	a=2, b=1	(203, 52, 123, 1020, 120, 491, 2113, 460, 3)	7189480.11	1949794.0	8933.71	0.989
	a=1, b=1	(210, 51, 137, 1004, 121, 500, 2122, 464, 77)	3751110.714	1974674	8984.42	0.988
	a=0.5, b=1	(210, 69, 132, 1012, 124, 475, 2140, 467, 28)	2721162.081	2031868.5	9245.14	0.979
	a=0.1, b=1	(210, 70, 138, 1016, 124, 498, 2135, 476, 97)	2164376.152	2046334.5	9355.49	0.976
	a=2, b=1	(200, 51, 133, 1003, 121, 473, 1982, 462, 9)	7272507.75	1887932.0	8829.36	0.983
	a=1, b=1	(210, 52, 131, 1029, 122, 447, 2103, 468, 31)	3688574.06	1965712.5	9041.35	0.976
	a=0.5, b=1	(208, 69, 120, 1000, 123, 480, 2138, 464, 36)	2723908.596	2017709.5	9174.00	0.972
	a=0.1, b=1	(209, 70, 140, 1027, 120, 488, 2136, 461, 199)	2167596.39	2047318.5	9356.06	0.971
	a=2, b=1	(210, 50, 132, 1000, 120, 411, 1999, 460, 15)	7378074.89	1901368.0	8811.67	0.982
	a=1, b=1	(208, 51, 135, 1001, 120, 476, 2132, 462, 29)	3777555.73	1967299.5	8933.87	0.975
	a=0.5, b=1	(209, 70, 139, 1002, 120, 434, 2132, 464, 38)	2734517.49	2024983.0	9171.42	0.971
	a=0.1, b=1	(208, 69, 134, 1114, 131, 483, 2134, 460, 112)	2174239.97	2058580.0	9553.91	0.968

The above analysis and computation results indicate: 1) BSE-SGM has good structural characteristics; 2) For the same stochastic programming problem, the variations of the decision results and the satisfaction probability of constraints by using BSE-SGM are generally smaller than that of expectation model, which shows that the decision reliability of BSE-SGM is greater than that of the expectation model; 3) For the different synthesis effect functions, the decision results are different, and even the difference is great. Therefore, BSE-SGM can effectively integrate uncertainty process consciousness into decision process.

6. **Conclusion.** In this paper, for the solution of stochastic programming, by analyzing the deficiencies of the existing methods, combining with the essential features of stochastic decision, we give the comparison method based on synthesis effect for random information, and give an axiomatic system for random synthesis effect function, and establish a general solution model for stochastic programming problem; further, we analyze the feature of our model through an example. All the results indicate that BSE-SGM has good structural characteristics, it not only contains the existing random programming methods, and can integrate the processing consciousness of stochastic information into the quantitative operation process, and it extends and enriches the existing relevant theories, which laid the foundation for further establishing random optimization theories and methods.

Acknowledgment. This work is supported by the National Natural Science Foundation of China (71071049) and the Natural Science Foundation of Hebei Province (F2011208056).

REFERENCES

[1] B. D. Liu, R. Q. Zhao and G. Wang, *Uncertain Programming and Application*, Tsinghua University Press, Beijing, 2003.

[2] A. Charnes and W. W. Cooper, *Management Models and Industrial Applications of Linear Programming*, Johhn Wiley and Sons, Inc, New York, 1961.

[3] B. Liu, Dependent-chance programming: A class of stochastic programming, *Computers and Mathematics with Applications*, vol.34, no.12, pp.89-104, 1997.

[4] J. K. Song, Z. X. Zhang and Y. Zhang, A stochastic expected value model for measures program of oil field, *Journal of Shandong University of Technology (Sci. and Tech.)*, vol.20, pp.9-12, 2006.

[5] L. X. Yang and Y. Feng, Abicriteria solid transportation problem with fixed charge under stochastic environment, *Applied Mathematical Modelling*, vol.17, pp.2668-2683, 2007.

[6] T. W. Chien, Dertermining profit-maximizing product/shipping policies in a one-to-one direct shipping, stochastic demand environment, *European Journal of Operational Research*, vol.64, pp.83-102, 1993.

[7] Y. J. Sun, L. Y. Kang and W. X. Shi, Study on sizing of batteries for distributed power system utilizing chance constrained programming, *Journal of System Simulation*, vol.17, no.1, pp.41-44, 2005.

[8] M. S. Chang, Y. L. Tseng and J. W. Chen, A scenario planning approach for the flood emergency logistics preparation problem under uncertainty, *Transportation Research Part E*, vol.43, pp.737-754, 2007.

[9] X. D. Ding, R. Q. Wu and S. H. Shao, Hybrid programming model with fuzzy and stochastic parameters, *Control and Decision*, vol.17, pp.587-590, 2002.

[10] K. Iwamura and B. D. Liu, A genetic algorithm for chance constrained programming, *Journal of Information and Optimization Sciences*, vol.17, no.2, pp.40-47, 1996.

[11] R. Zhao, K. Iwamura and B. D. Liu, Chance constrained integer programming and stochastic simulation based genetic algorithms, *Journal of Systems Science and Systems Engineering*, vol.7, no.1, pp.96-102, 1998.

[12] B. D. Liu, Dependent-chance goal programming and its genetic algorithm based approach, *Mathematical and Computer Modelling*, vol.24, no.7, pp.43-52, 1996.

[13] N. Xiao, J. C. Zeng and W. B. Li, Solving stochastic expected value models with hybrid intelligent algorithm of based on PSO, *Computer Engineering and Applications*, vol.45, no.10, pp.45-48, 2009.

[14] F. Li, C. Jin and Y. Shi, Study on solution models and methods for random assignment problem, *International Journal of Innovative Computing, Information and Control*, vol.5, no.2, pp.359-368, 2009.

[15] H. M. Li, J. Sun and W. B. Xu, Empirical study based on quantum-behaved particle swarm optimization stochastic programming algorithm, *Computer Engineering and Applications*, vol.43, pp.185-188, 2007.

[16] B. D. Liu and R. Zhao, *Fuzzy Programming and Stochastic Programming*, Tsinghua University Press, Beijing, 2004.