

# Risk Allocation Strategies for Distributed Chance-Constrained Task Allocation

Sameera S. Ponda, Luke B. Johnson and Jonathan P. How  
Laboratory for Information and Decision Systems, MIT

**Abstract**—This paper addresses the issue of allocating risk amongst agents in distributed chance-constrained planning algorithms. Building on previous research that extended chance-constrained planning to stochastic multi-agent multi-task missions, this paper presents a framework for risk allocation and proposes several strategies for distributing risk in homogeneous and heterogeneous teams. In particular, the contributions of this work include proposing risk allocation strategies that exploit domain knowledge of agent score distributions to improve team performance, and providing insights about what stochastic parameters and features affect the allocations and the overall mission score/performance. Results are provided showing improved performance over previously published heuristic techniques in environments with given allowable risk thresholds.

## I. INTRODUCTION

The use of autonomous systems, such as unmanned aerial and ground vehicles (UAVs/UGVs), has motivated the development of autonomous task allocation and planning methods that ensure spatial and temporal coordination for teams of cooperating agents. The basic planning problem can be formulated as a combinatorial optimization, often involving nonlinear and time-varying system dynamics. For most problems of interest, optimal solution methods are computationally intractable and approximation techniques are regularly employed [1]. Most of these consist of centralized planning approaches, which usually require high bandwidth communications and react slowly to local changes in dynamic environments, motivating the development of distributed algorithms where agents plan individually and coordinate with each other locally [2]. One class of distributed planning algorithms involves using auction algorithms augmented with consensus protocols, which are particularly well suited to developing real-time conflict-free solutions in dynamic environments [3]–[5].

An important issue associated with autonomous planning is that the algorithms rely on underlying system models and parameters which are often subject to uncertainty. This uncertainty can result from many sources including: inaccurate parameters, model simplifications, fundamentally nondeterministic processes (e.g. sensor readings, stochastic dynamics), and dynamic local information changes. As discrepancies between planner models and actual system dynamics increase, mission performance typically degrades. Furthermore, the impact of these discrepancies on the overall plan quality is usually hard to quantify in advance due to

nonlinear effects, coupling between tasks and agents, and interdependencies between system constraints (e.g. longer-than-expected service times impact the arrival times of subsequent tasks). However, if uncertainty models of planning parameters are available, they can be leveraged to create robust plans that explicitly hedge against the inherent uncertainty given allowable risk thresholds.

Several stochastic planning strategies have been considered throughout the literature employing various stochastic metrics (e.g. expected value, worst-case performance, CVAR) [6]–[10]. A particular stochastic metric that can be used when low probability of failure is required is the chance-constrained metric [8,11,12], which provides probabilistic guarantees on achievable mission performance given allowable risk thresholds. This work builds upon our previous efforts to extend chance-constrained planning to distributed environments [13]. In particular, this paper proposes a formal approach to allocating the risk among the agents, and derives several risk allocation strategies for homogeneous and heterogeneous teams of agents that can be used within a distributed chance-constrained planning framework.

## II. PROBLEM STATEMENT

This section describes the robust task allocation problem formulation and the chance-constrained stochastic metric for distributed planning applications.

### A. Robust Distributed Planning

Given a list of  $N_a$  agents and  $N_t$  tasks, the robust task assignment problem can be written as follows:

$$\begin{aligned} \max_{\mathbf{x}} \quad & \mathbb{M}_{\theta} \left\{ \sum_{i=1}^{N_a} \left( \sum_{j=1}^{N_t} c_{ij}(\mathbf{x}_i, \boldsymbol{\tau}_i, \boldsymbol{\theta}) \right) \right\} \\ \text{s.t.} \quad & \sum_{i=1}^{N_a} x_{ij} \leq 1, \quad \forall j \in \{1, \dots, N_t\} \\ & \mathbf{x} \in \{0, 1\}^{N_a \times N_t}, \quad \boldsymbol{\tau} \in \{\mathbb{R}^+ \cup \emptyset\}^{N_a \times N_t} \end{aligned} \quad (1)$$

where  $\mathbf{x}$ , is a set of binary decision variables,  $x_{ij}$ , which are used to indicate whether or not task  $j$  is assigned to agent  $i$ ;  $c_{ij}$  is the reward agent  $i$  receives for task  $j$  given the agent's overall assignment and parameters;  $\boldsymbol{\tau}$  represents a set of task execution times;  $\boldsymbol{\theta}$  is the set of stochastic planning parameters, with joint distribution  $\mathbf{f}_{\theta}(\boldsymbol{\theta})$ , that affect the score calculation; and finally  $\mathbb{M}_{\theta}(\cdot)$  represents some stochastic metric acting upon the overall mission score subject to the uncertainty in  $\boldsymbol{\theta}$ . The objective of Eq. (1) is to find a conflict-free allocation of tasks to agents (no more than one agent per

S. Ponda and L. Johnson are with the Dept. of Aeronautics and Astronautics, MIT, Cambridge, MA, {sponda, lbj16}@mit.edu

J. P. How is the R. C. Maclaurin Professor of Aeronautics and Astronautics, MIT, Cambridge, MA, jhow@mit.edu

task, as specified by the constraint in Eq. (1)), that maximizes the stochastic reward  $\mathbb{M}_\theta(\cdot)$ .

If the metric  $\mathbb{M}_\theta(\cdot)$  allows the sum over agents to be extracted, then a distributed version of Eq. (1) can be written as follows,

$$\begin{aligned} \max_{\mathbf{x}_i} \quad & \mathbb{M}_\theta \left\{ \sum_{j=1}^{N_t} \mathbf{c}_{ij}(\mathbf{x}_i, \boldsymbol{\tau}_i, \boldsymbol{\theta}) \right\} \\ \text{s.t.} \quad & \sum_{i=1}^{N_a} x_{ij} \leq 1, \quad \forall j \in \{1, \dots, N_t\} \\ & \mathbf{x}_i \in \{0, 1\}^{N_t}, \quad \boldsymbol{\tau}_i \in \{\mathbb{R}^+ \cup \emptyset\}^{N_t} \end{aligned} \quad (2)$$

where each agent  $i$  optimizes its own assignment  $\mathbf{x}_i$  subject to its own stochastic score  $\mathbb{M}_\theta(\cdot)$ , and the only coupling between agents is given by the conflict-free constraint (which involves a sum over agents, as shown in Eq. (2)). Given this distributed framework, robust distributed algorithms can be developed to perform the task allocation in these stochastic environments [5].

### B. Chance-Constrained Distributed Planning

Of interest in this work is the chance-constrained stochastic metric which provides probabilistic guarantees on achievable mission performance given allowable risk thresholds [8, 11,12]. This metric is useful for missions where low probability of failure is required. Substituting  $\mathbb{M}_\theta(\cdot)$  in Eq. (1) with a chance-constrained metric, the objective function becomes,

$$\begin{aligned} \max_{\mathbf{x}} \quad & \mathbf{y} \\ \text{s.t.} \quad & \mathbb{P}_\theta \left\{ \sum_{i=1}^{N_a} \left( \sum_{j=1}^{N_t} \mathbf{c}_{ij}(\mathbf{x}_i, \boldsymbol{\tau}_i, \boldsymbol{\theta}) \right) \leq \mathbf{y} \right\} \leq \epsilon \end{aligned} \quad (3)$$

Unfortunately, Eq. (3) introduces an additional constraint to the optimization that couples agents through a probabilistic chance constraint. Using this metric, the sum over agents cannot be easily extracted from the optimization, making distributed implementations nontrivial [3,5]. In previous work [13], we proposed a distributed chance-constrained approximation of the form,

$$\begin{aligned} \max_{\mathbf{x}_i} \quad & \mathbf{y}_i \\ \text{s.t.} \quad & \mathbb{P}_\theta \left\{ \left( \sum_{j=1}^{N_t} \mathbf{c}_{ij}(\mathbf{x}_i, \boldsymbol{\tau}_i, \boldsymbol{\theta}) \right) \leq \mathbf{y}_i \right\} \leq \epsilon_i \end{aligned} \quad (4)$$

where each agent  $i$  solved its own chance-constrained optimization subject to an individual risk threshold  $\epsilon_i$ . The approximate chance-constrained mission score was then given by the sum over these chance-constrained agent scores, and an equivalence between the two problem formulations was obtained if the individual agent risks  $\epsilon_i$  satisfied the constraint,

$$\sum_{i=1}^{N_a} \mathbf{F}_{\mathbf{z}_i}^{-1}(\epsilon_i) = \mathbf{F}_{\mathbf{z}}^{-1}(\epsilon) \quad (5)$$

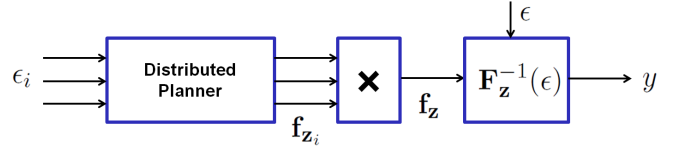


Fig. 1. Relationship between agent risks and chance-constrained score in distributed chance-constrained planning. The main pieces include the distributed planner, which uses the risk allocations to make agent plans, a convolution block that combines the agent score distributions associated with the agent plans to derive the mission score distribution, and a final block that computes the chance-constrained mission score given the score distribution and the allowable mission risk threshold.

where  $\epsilon$  is the mission risk,  $\epsilon_i, \forall i$  are the individual agent risks,  $\mathbf{z}_i = \sum_{j=1}^{N_t} \mathbf{c}_{ij}(\mathbf{x}_i, \boldsymbol{\tau}_i, \boldsymbol{\theta}), \forall i$  are the random agent scores subject to the uncertainty in  $\boldsymbol{\theta}$  (with distributions  $\mathbf{f}_{\mathbf{z}_i}(\mathbf{z}_i)$  and CDFs  $\mathbf{F}_{\mathbf{z}_i}(\cdot), \forall i$ ), and  $\mathbf{z} = \sum_{i=1}^{N_a} \mathbf{z}_i$  is the random mission score (with distribution  $\mathbf{f}_{\mathbf{z}}(\mathbf{z})$  and CDF  $\mathbf{F}_{\mathbf{z}}(\cdot)$ ). Eq. (5) identifies the relationship between mission risk and agent risks given available distributions for both agent and mission scores, however, in chance-constrained planning problems it is difficult to predict what these distributions will be a priori.

There are two main goals associated with allocating the risks  $\epsilon_i$  amongst the agents: (1) to ensure that the global mission risk level is adequately captured (which was the purpose of Eq. (5)), and (2) to find risk allocations that encourage agents to pick “better” plans, such that the chance-constrained mission score  $\mathbf{y} = \mathbf{F}_{\mathbf{z}}^{-1}(\epsilon)$  is maximized. This involves finding a distribution for the mission score  $\mathbf{z}$  that maximizes  $\mathbf{F}_{\mathbf{z}}^{-1}(\epsilon)$  given an allowable risk threshold  $\epsilon$ . However, as illustrated in Fig. 1,  $\mathbf{f}_{\mathbf{z}}(\mathbf{z})$  is a function of the agent score distributions  $\mathbf{f}_{\mathbf{z}_i}(\mathbf{z}_i)$  (e.g. convolution if agents are independent), and the distributions  $\mathbf{f}_{\mathbf{z}_i}(\mathbf{z}_i)$  are in turn functions of the risk levels  $\epsilon_i$  and of the inner workings of the planner, which are hard to predict. This severe coupling makes the task of optimizing risk allocations  $\epsilon_i$  in order to achieve the best plan very difficult. This work presents a formal approach to risk allocation and proposes several risk allocation strategies for both homogeneous and heterogeneous teams. Results are provided comparing the performance of these strategies, and insights are provided into what features affect the chance-constrained team performance in distributed environments.

### III. RISK ALLOCATION STRATEGIES

Previous work presented a heuristic risk allocation strategy based on simplified assumptions – Gaussian distributions and identical agent risk allocations – to set the agent risks  $\epsilon_i$  given a team of heterogeneous agents [13]. This paper presents a more formal approach to risk allocation, and proposes several new risk allocation strategies for both homogeneous and heterogeneous teams of agents. In particular, the contributions of this work include: (1) presenting a general framework for homogeneous and heterogeneous risk allocation, (2) exploiting domain knowledge of agent score distributions to improve risk allocation for homogeneous teams, (3) proposing risk allocation strategies for heterogeneous teams based

on knowledge of the different agent types (where agent risks are allowed to be nonidentical), and (4) providing insights, for both homogeneous and heterogeneous cases, on how to compare the different risk allocation strategies, what parameters affect these allocations, and what features affect the performance of the overall chance-constrained mission score given the distributed approximation of Eq. (4).

The remainder of this section provides details on the different risk allocation strategies based on Eq. (5) for teams of homogeneous and heterogeneous agents, and comparisons of team performance given these risk allocation strategies are provided in Section V.

### A. Homogeneous Risk Allocation

The first case considered is for teams of homogeneous agents, where all  $N_a$  agents are assumed to have identical score distributions  $\mathbf{f}_{\mathbf{z}_i}(\mathbf{z}_i)$ , and identical risk allocations  $\epsilon_i$ . Using these assumptions in Eq. (5) gives the general expression for homogeneous risk allocation,

$$\epsilon_i = \mathbf{F}_{\mathbf{z}_i} \left( \frac{1}{N_a} \mathbf{F}_{\mathbf{z}_i}^{-1}(\epsilon) \right) \quad (6)$$

Specifying the mission distribution  $\mathbf{f}_{\mathbf{z}}(\mathbf{z})$  involves convolving the  $N_a$  agent score distributions, which is often analytically intractable. This work uses a Gaussian distribution to approximate the mission score invoking the Central Limit Theorem (sums of random variables can be approximated as Gaussian), where  $\mathbf{z} \sim \mathcal{N}(\mu, \sigma^2)$  with  $\mu = N_a \mu_i$  and  $\sigma^2 = N_a \sigma_i^2$ . For Gaussian random variables, the CDF and inverse CDF are given by,

$$\begin{aligned} \mathbf{F}_X(x) &= \frac{1}{2} + \frac{1}{2} \operatorname{erf} \left( \frac{x - \mu}{\sqrt{2\sigma^2}} \right) \\ \mathbf{F}_X^{-1}(\epsilon) &= \mu + \sqrt{2\sigma^2} \operatorname{erf}^{-1}(2\epsilon - 1) \end{aligned} \quad (7)$$

and using these expressions in Eq. (6) gives,

$$\epsilon_i = \mathbf{F}_{\mathbf{z}_i} \left( \mu_i + \sqrt{\frac{2}{N_a}} \sigma_i \operatorname{erf}^{-1}(2\epsilon - 1) \right) \quad (8)$$

Eq. (8) can be used with many different forms of the agent distributions  $\mathbf{f}_{\mathbf{z}_i}(\mathbf{z}_i)$ . This work explores three different distributions for homogeneous agents based on Gaussian, exponential and gamma distributions. Fig. 2 illustrates the shapes of the distributions used in these strategies and derivation details are provided in the following sections.

1) *Gaussian Risk Allocation*: In this risk allocation strategy, the agent scores are assumed to be Gaussian random variables  $\mathbf{z}_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$  with mean  $\mu_i$  and variance  $\sigma_i^2$ . Replacing  $\mathbf{F}_{\mathbf{z}_i}(\cdot)$  in Eq. (8) with a Gaussian CDF (Eq. (7)) gives the following derivation for the agent risks,

$$\begin{aligned} \epsilon_i &= \frac{1}{2} + \frac{1}{2} \operatorname{erf} \left( \frac{\mu_i + \sqrt{\frac{2}{N_a}} \sigma_i \operatorname{erf}^{-1}(2\epsilon - 1) - \mu_i}{\sqrt{2\sigma_i^2}} \right) \\ &= \frac{1}{2} \left( 1 + \operatorname{erf} \left( \sqrt{\frac{1}{N_a}} \operatorname{erf}^{-1}(2\epsilon - 1) \right) \right) \end{aligned}$$

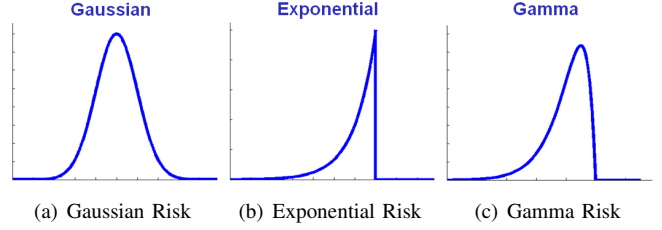


Fig. 2. Agent score distributions for three different homogeneous risk allocation strategies.

This Gaussian risk allocation will only be accurate when agent scores approach Gaussian distributions, however, several scenarios of interest involve nonsymmetric agent distributions. For example, the scenarios explored in this work consist of time-critical missions, where arriving at tasks early or on time results in maximum task rewards, and arriving late yields diminishing rewards. As a result, agent score distributions tend to have higher probability masses concentrated around sums of maximum task rewards, and diminishing tails towards lower scores, motivating the use of nonsymmetric agent distributions within the risk allocation. This work uses gamma and exponential distributions (flipped about the vertical axis and shifted), as shown in Fig. 2.

2) *Exponential Risk Allocation*: Flipping about the vertical axis and shifting distributions involves applying a linear transformation,  $Y = aX + b$ , where  $a = -1$  (flip) and  $b$  is some quantity corresponding to the shift. For these linear transformations of random variables, the CDF of the transformed random variable  $\mathbf{F}_Y(y)$  can be computed given the original CDF  $\mathbf{F}_X(x)$  as follows,

$$\mathbf{F}_Y(y) = \begin{cases} \mathbf{F}_X \left( \frac{y-b}{a} \right), & a > 0 \\ 1 - \mathbf{F}_X \left( \frac{y-b}{a} \right), & a < 0 \end{cases} \quad (9)$$

For an exponential distribution with parameter  $\lambda$ , the CDF, mean, and variance of the original and transformed distributions are given by,

$$\begin{aligned} \mathbf{F}_X(x) = 1 - e^{-\lambda x} &\quad \mathbf{F}_Y(y) = e^{-\lambda(b-y)} \\ \mu_X = \frac{1}{\lambda} &\quad \Rightarrow \quad \mu_Y = \frac{1}{\lambda} + b \\ \sigma_X^2 = \frac{1}{\lambda^2} &\quad \sigma_Y^2 = \frac{1}{\lambda^2} \end{aligned}$$

which can be used in Eq. (8) to derive the following expression for agent risks,

$$\begin{aligned} \epsilon_i &= e^{-\lambda \left( b - \mu_i - \sqrt{\frac{2}{N_a}} \sigma_i \operatorname{erf}^{-1}(2\epsilon - 1) \right)} \\ &= e^{-\lambda \left( b + \frac{1}{\lambda} - b - \sqrt{\frac{2}{N_a}} \left( \frac{1}{\lambda} \right) \operatorname{erf}^{-1}(2\epsilon - 1) \right)} \\ &= e^{-\left( 1 - \sqrt{\frac{2}{N_a}} \operatorname{erf}^{-1}(2\epsilon - 1) \right)} \end{aligned}$$

Although this exponential-based distribution is nonsymmetric and captures properties of agent scores better than the Gaussian risk allocation, the shape of the distribution is fixed (the *scale* can be controlled through the parameter  $\lambda$  but the *shape* is fixed). In some situations, it is preferable to use a gamma distribution instead, since it provides more control over the shape of the distribution as well as the scale through

the additional parameter  $k$ .

3) *Gamma Risk Allocation*: For this risk allocation, a linear transformation with  $a = -1$  and shift value  $b$  is again applied, this time to the gamma distribution. Eq. (9) with  $a < 0$  can again be used to derive the transformed CDF, and for a gamma distribution with parameters  $k$  and  $\theta$  (controlling the shape and scale respectively), the CDF, mean, and variance of the original and transformed random variables are given by,

$$\begin{aligned} \mathbf{F}_X(x) = \frac{1}{\Gamma(k)}\gamma\left(k, \frac{x}{\theta}\right) & \quad \mathbf{F}_Y(y) = 1 - \frac{1}{\Gamma(k)}\gamma\left(k, \frac{b-y}{\theta}\right) \\ \mu_X = k\theta & \quad \Rightarrow \quad \mu_Y = -k\theta + b \\ \sigma_X^2 = k\theta^2 & \quad \sigma_Y^2 = k\theta^2 \end{aligned}$$

where  $\Gamma(k) = \int_0^\infty e^{-t}t^{k-1}dt$  is the gamma function and  $\gamma(k, x) = \int_0^x e^{-t}t^{k-1}dt$  is the incomplete gamma function.

These can be used in Eq. (8) to obtain the following expression for agent risks,

$$\begin{aligned} \epsilon_i &= 1 - \frac{1}{\Gamma(k)}\gamma\left(k, \frac{1}{\theta}\left(b - \mu_i - \sqrt{\frac{2}{N_a}}\sigma_i \operatorname{erf}^{-1}(2\epsilon - 1)\right)\right) \\ &= 1 - \frac{1}{\Gamma(k)}\gamma\left(k, \frac{1}{\theta}\left(b + k\theta - b - \sqrt{\frac{2k}{N_a}}\theta \operatorname{erf}^{-1}(2\epsilon - 1)\right)\right) \\ &= 1 - \frac{1}{\Gamma(k)}\gamma\left(k, k - \sqrt{\frac{2k}{N_a}}\operatorname{erf}^{-1}(2\epsilon - 1)\right) \end{aligned}$$

In the case where  $k = 1$  the gamma distribution and the exponential distribution are equivalent (where  $\theta$  is related to  $\lambda$  by  $\theta = 1/\lambda$ ), and thus the gamma and exponential risk allocation strategies return the same values for  $\epsilon_i$ .

The risk expressions for the three homogeneous risk allocation strategies presented here are summarized as

$$\begin{aligned} \text{Gaussian} \quad \epsilon_i &= \frac{1}{2}\left(1 + \operatorname{erf}\left(\sqrt{\frac{1}{N_a}}\operatorname{erf}^{-1}(2\epsilon - 1)\right)\right) \\ \text{Exponential} \quad \epsilon_i &= e^{-\left(1 - \sqrt{\frac{2}{N_a}}\operatorname{erf}^{-1}(2\epsilon - 1)\right)} \\ \text{Gamma} \quad \epsilon_i &= 1 - \frac{1}{\Gamma(k)}\gamma\left(k, k - \sqrt{\frac{2k}{N_a}}\operatorname{erf}^{-1}(2\epsilon - 1)\right) \end{aligned} \quad (10)$$

Note that in all of these homogeneous risk allocation expressions, the agent risk values are not affected by the shift and scale parameters of the distributions (e.g.  $\mu$  and  $\sigma$  in the Gaussian case,  $b$  and  $\lambda$  in the exponential case, and  $b$  and  $\theta$  in the gamma case). The risk allocation remains constant regardless of the means and variances of the underlying distributions. Since the mission distribution is a function of the agent distributions, means and variances appear on both sides of Eq. (5) in equal magnitudes and thus cancel out. The intuition behind this observation is that the risk allocation process is affected by the shape of the distributions (particularly the tails), and not the scale and shift parameters or the distributions themselves.

### B. Heterogeneous Risk Allocation

Setting risk values for heterogeneous agents is more complicated, since the assumptions made in Eq. (6) may no longer hold (i.e. identical distributions, identical risks). In general, there are infinite possible combinations of  $\epsilon_i$  that are

valid solutions to Eq. (5) given a specific value of  $\epsilon$ , therefore specifying different individual agent risks is nontrivial. In previous work [13], we presented a heuristic risk allocation strategy for heterogeneous agents that assumed Gaussian distributions and assumed that all agent risk allocations were equal (same  $\epsilon_i, \forall i$ ). For this risk allocation strategy, using Gaussian distributions in Eq. (5) gave,

$$\begin{aligned} \sum_{i=1}^{N_a} \left( \mu_i + \sqrt{2\sigma_i^2} \operatorname{erf}^{-1}(2\epsilon_i - 1) \right) & \quad (11) \\ &= \mu + \sqrt{2\sigma^2} \operatorname{erf}^{-1}(2\epsilon - 1) \end{aligned}$$

where  $\mu = \sum_{i=1}^{N_a} \mu_i$  and  $\sigma^2 = \sum_{i=1}^{N_a} \sigma_i^2$ , and assuming equal risks  $\epsilon_i$  for all agents gave,

$$\epsilon_i = \frac{1}{2} \left( 1 + \operatorname{erf}(\mathbf{H} \operatorname{erf}^{-1}(2\epsilon - 1)) \right) \quad (12)$$

$$\mathbf{H} = \left( \frac{\sqrt{\sum_{i=1}^{N_a} \sigma_i^2}}{\sum_{i=1}^{N_a} \sigma_i} \right)$$

where the constant value  $\mathbf{H}$  represented team heterogeneity with regards to variance in agents' scores. With homogeneous agents,  $\mathbf{H} = \sqrt{1/N_a}$  and Eq. (12) is equivalent to the homogeneous Gaussian risk allocation of Eq. (10). On the other hand, if the variance of the mission distribution was only due to one agent (all other agents are deterministic with no variance), then  $\mathbf{H} = 1$  and  $\epsilon_i = \epsilon$  as expected. Thus selecting  $\mathbf{H} \in \left[ \frac{1}{\sqrt{N_a}}, 1 \right]$  was used in [13] as a way to represent team heterogeneity.

This work proposes a more general risk allocation strategy for heterogeneous teams, which involves assigning different risk values  $\epsilon_i$  to different types of agents. This strategy assumes that the agents can be grouped into  $K$  types, where agents in group  $k$  have the same distribution  $\mathbf{f}_{\mathbf{z}_k}(\mathbf{z}_k)$ , and are assigned identical risk allocations  $\epsilon_k$ . Using these assumptions, Eq. (5) becomes

$$\sum_{k=1}^K N_k \mathbf{F}_{\mathbf{z}_k}^{-1}(\epsilon_k) = \mathbf{F}_{\mathbf{z}}^{-1}(\epsilon) \quad (13)$$

where  $N_k$  is the number of agents of type  $k \in \{1, \dots, K\}$ . Even though Eq. (13) is simpler than Eq. (5), it still involves performing computations that are analytically intractable for most distribution types. The approach taken in this work again leverages the Central Limit Theorem and assumes a Gaussian distribution for the mission score, as well as assuming Gaussian distributions for agent scores to maintain analytic tractability. Given Gaussian distributions, Eq. (13) becomes

$$\begin{aligned} \sum_{k=1}^K N_k \left( \mu_k + \sqrt{2\sigma_k^2} \operatorname{erf}^{-1}(2\epsilon_k - 1) \right) & \quad (14) \\ &= \mu + \sqrt{2\sigma^2} \operatorname{erf}^{-1}(2\epsilon - 1) \end{aligned}$$

where  $\mu = \sum_{k=1}^K N_k \mu_k$  and  $\sigma^2 = \sum_{k=1}^K N_k \sigma_k^2$ , which

simplifies to

$$\sum_{k=1}^K N_k \sigma_k \operatorname{erf}^{-1}(2\epsilon_k - 1) = \sigma \operatorname{erf}^{-1}(2\epsilon - 1) \quad (15)$$

An issue that still remains is that Eq. (15) involves solving for  $K$  unknown variables  $\epsilon_k$ , for which there are infinite possible solutions. To address this issue, the right hand side of Eq. (15) can be divided into shares  $m_k$  (where  $\sum_{k=1}^K m_k = 1$ ), giving the following equation for each agent type  $k$ ,

$$N_k \sigma_k \operatorname{erf}^{-1}(2\epsilon_k - 1) = m_k \sigma \operatorname{erf}^{-1}(2\epsilon - 1)$$

which can be used to solve for the risk allocation for each agent group  $k$ ,

$$\begin{aligned} \epsilon_k &= \frac{1}{2} \left( 1 + \operatorname{erf}(\mathbf{H}_k \operatorname{erf}^{-1}(2\epsilon - 1)) \right), \quad \forall k \quad (16) \\ \mathbf{H}_k &= \left( \frac{m_k \sigma}{N_k \sigma_k} \right) \end{aligned}$$

Using Eq. (16) typically gives different risk allocations  $\epsilon_k$  for the different agent types  $k$ , depending on the values of  $\mathbf{H}_k$ , which are functions of the selected shares  $m_k$  and the statistics of each agent group. It is not immediately obvious how to partition these shares to get risk allocations for the team that optimize the chance-constrained mission score (due to the complexity alluded to in Fig. 1). This work explores a few different strategies to allocate the shares  $m_k$  which are described next.

1) *Equal Shares*: Assuming equal shares for all agents gives  $m_k \propto N_k$ , which can be substituted into Eq. (16) to give,

$$m_k = \frac{N_k}{\sum_{k=1}^K N_k} \Rightarrow \mathbf{H}_k = \frac{\sigma}{N_a \sigma_k} \quad (17)$$

The first term of  $\mathbf{H}_k$  involves statistics affecting all agents equally, whereas the second term is affected by  $\sigma_k$  for agents of type  $k$ . As a result, the risk allocations  $\epsilon_k$  for the different agent types will typically be different.

2) *Shares Proportional to Variance*: The second case assumes that the shares are proportional to the group variance,  $m_k \propto N_k \sigma_k^2$ , where substituting into Eq. (16) gives,

$$m_k = \frac{N_k \sigma_k^2}{\sum_{k=1}^K N_k \sigma_k^2} \Rightarrow \mathbf{H}_k = \frac{1}{\sigma} \sigma_k \quad (18)$$

Again the second term is affected by  $\sigma_k$ , so the risk allocations  $\epsilon_k$  for the different agent types will typically differ.

3) *Shares Proportional to Standard Deviation*: The last strategy assumes that the shares are proportional to the group's standard deviation,  $m_k \propto \sqrt{N_k \sigma_k^2}$ , where  $\mathbf{H}_k$  becomes

$$m_k = \frac{\sqrt{N_k \sigma_k^2}}{\sum_{k=1}^K \sqrt{N_k \sigma_k^2}} \Rightarrow \mathbf{H}_k = \frac{\sigma}{\sum_{k=1}^K \sqrt{N_k \sigma_k^2}} \frac{1}{\sqrt{N_k}}$$

In this special case, the agent risk allocations are not affected by each group's variance. The first term of  $\mathbf{H}_k$  is constant for all agents, and the second term is only affected by the

number of agents in the group. In particular, if each group has the same number of agents, the risk allocations for all agents will be identical, even if the agents have different variances. Furthermore, taking each agent to be its own type (i.e.  $N_k = 1, \forall k$ , and  $K = N_a$ ), the expression for  $\mathbf{H}_k$  is equivalent to  $\mathbf{H}$  in Eq. (12), therefore the risk allocation strategy proposed in [13] is a specific case of this more general risk allocation framework where shares are proportional to agent standard deviation.

Note that, in all of the above heterogeneous risk allocation strategies, the agent risks do not depend on the means of the agent score distributions or mission score distribution, they only depend on the variances. This is similar to the observation made for the homogeneous risk allocation strategies, where the means of the distributions and scale parameters did not affect the risk allocation. However, in the heterogeneous case, the *relative* scale parameters (normalized variances) do affect the risk allocation, as captured by the constants  $\mathbf{H}_k$  which are functions of  $\sigma_k$ .

#### IV. DISTRIBUTED CHANCE-CONSTRAINED ALGORITHMIC FRAMEWORK

Once risks have been allocated amongst the different agents, a distributed algorithm can be used to solve Eq. (4), where each agent  $i$  solves its own chance-constrained optimization subject to its individual risk threshold  $\epsilon_i$ , and deconflicts assignments with other agents through local communications. For this purpose, this work uses a robust extension to the Consensus-Based Bundle Algorithm (CBBA), a polynomial-time distributed auction algorithm that provides provably good real-time approximate solutions for multi-agent multi-task allocation problems [3]. CBBA consists of iterations between two phases: a bundle building phase where each agent greedily selects a set of tasks to execute, and a consensus phase where conflicting assignments are resolved through local communications with neighboring agents. In recent work, CBBA was extended to account for stochastic metrics within the distributed planning framework [5]. The Robust CBBA algorithm is summarized in Alg. 1. The data structures in  $\mathcal{A}_i$  consist of internal decisions made by each agent (including a path sorted by task execution order  $\mathbf{p}_i$ , and a task bundle sorted in order of task assignment  $\mathbf{b}_i$ ). The data structures in  $\mathcal{C}_i$  involve data to be communicated between agents (i.e. a winning agent list  $\mathbf{z}_i$ , the corresponding winning bid list  $\mathbf{y}_i$ , and a set of communication timestamps  $\mathbf{t}_i$  (for more details see [3]–[5])).

Alg. 2 describes the process by which agents decide to add tasks to their current bundle  $\mathbf{b}_i$ . A detailed description is available in [5], but the main algorithmic steps are as follows: for each available task  $j$ , compute the best stochastic score for adding the task to the current path  $\mathbf{p}_i$ , along with the corresponding best location  $n_j^*$  for task  $j$  in the path (line 3); compute the improvement in score and corresponding bid information for communication with other agents (lines 4-6, see [4] for details); select the task that leads to the largest improvement in score (line 8), and add it to the current set of assignments (lines 10-13); repeat until no profitable tasks

---

**Algorithm 1** CBBA( $\mathcal{I}, \mathcal{J}$ )

---

```
1: Initialize  $\{\mathcal{A}_i, \mathcal{C}_i\}, \forall i \in \mathcal{I}$ 
2: while  $\neg$ converged do
3:    $(\mathcal{A}_i, \mathcal{C}_i) \leftarrow$  CBBA-BUNDLE-ADD( $\mathcal{A}_i, \mathcal{C}_i, \mathcal{J}$ ),  $\forall i \in \mathcal{I}$ 
4:    $(\mathcal{A}_i, \mathcal{C}_i) \leftarrow$  CBBA-CONSENSUS( $\mathcal{A}_i, \mathcal{C}_i, \mathcal{C}_{N_i}$ ),  $\forall i \in \mathcal{I}$ 
5:   converged  $\leftarrow \bigwedge_{i \in \mathcal{I}}$  CHECK-CONVERGENCE( $\mathcal{A}_i$ )
6: end while
7:  $\mathcal{A} \leftarrow \bigcup_{i \in \mathcal{I}} \mathcal{A}_i$ 
8: return  $\mathcal{A}$ 
```

---

---

**Algorithm 2** CBBA-BUNDLE-ADD( $\mathcal{A}_i, \mathcal{C}_i, \mathcal{J}$ )

---

```
1: while  $|\mathbf{p}_i| \leq L_i$  do
2:   for  $j \in \mathcal{J} \setminus \mathbf{p}_i$  do
3:      $J_{(\mathbf{p}_i \oplus_{n_j} j)} \leftarrow \max_{n_j} \text{CC-PATH-SCORE}(\mathbf{p}_i \oplus_{n_j} j)$ 
4:      $\Delta J_{ij}(\mathbf{p}_i) = J_{(\mathbf{p}_i \oplus_{n_j} j)} - J_{\mathbf{p}_i}$ 
5:      $s_{ij} = \min(\Delta J_{ij}(\mathbf{p}_i), y_{ik}), \forall k \in \mathbf{p}_i$ 
6:      $h_{ij} = \mathbb{I}(s_{ij} > y_{ij})$ 
7:   end for
8:    $j^* = \operatorname{argmax}_{j \notin \mathbf{p}_i} \Delta J_{ij}(\mathbf{p}_i) h_{ij}$ 
9:   if  $(\Delta J_{ij^*}(\mathbf{p}_i) h_{ij^*} > 0)$  then
10:     $\mathbf{b}_i \leftarrow (\mathbf{b}_i \oplus_{\text{end}} j^*)$ 
11:     $\mathbf{p}_i \leftarrow (\mathbf{p}_i \oplus_{n_j} j^*)$ 
12:     $z_{ij^*} \leftarrow i$ 
13:     $y_{ij^*} \leftarrow s_{ij^*}$ 
14:   else
15:     break
16:   end if
17: end while
18:  $\mathcal{A}_i \leftarrow \{\mathbf{b}_i, \mathbf{p}_i\}$ 
19:  $\mathcal{C}_i \leftarrow \{\mathbf{z}_i, \mathbf{y}_i, \mathbf{t}_i\}$ 
20: return  $(\mathcal{A}_i, \mathcal{C}_i)$ 
```

---

are available or until a maximum path length  $L_i$  has been reached.

One complication with evaluating stochastic path scores (as required by Alg. 2, line 3), is that for every realization of the uncertain planning parameters, decisions about optimal task execution times need to be made. In general, given infinite support of the uncertain parameters, this would involve an uncountable number of optimizations. Alg. 3 presents a sampling approximation to the chance-constrained path score that can be used to maintain analytical tractability given this issue regarding optimization of task execution times. The main algorithmic steps involve: selecting  $N$  representative samples  $\theta_k$  with corresponding probabilistic weights  $w_k$  (lines 1-2); computing the optimal execution times  $\tau_i^*$  and corresponding path score  $J_{\mathbf{p}_i}^k$  for each sample value of the planning parameters  $\theta_k$  (lines 3-6); constructing a discrete approximation to the score distribution by sorting the score samples (with associated weights) in ascending order (line 7); and computing an approximate chance-constrained score given this discrete distribution and the allowable risk threshold  $\epsilon_i$  (line 8). The next section provides results comparing the different risk allocation strategies presented in this paper for setting the risks  $\epsilon_i$  given time-critical mission scenarios.

---

**Algorithm 3** CC-PATH-SCORE( $\mathbf{p}_i$ )

---

```
1:  $\{\theta_1, \dots, \theta_N\} \sim \mathbf{f}(\theta)$ 
2:  $\{w_1, \dots, w_N\} \leftarrow \{w_1, \dots, w_N\} / \sum_{k=1}^N w_k$ 
3: for  $k \in \{1, \dots, N\}$  do
4:    $\tau_i^* = \operatorname{argmax}_{\tau_i} \sum_{j=1}^{N_i} \mathbf{c}_{ij}(\tau_{ij}(\mathbf{p}_i), \theta_k) x_{ij}$ 
5:    $J_{\mathbf{p}_i}^k = \sum_{j=1}^{N_i} \mathbf{c}_{ij}(\tau_{ij}^*(\mathbf{p}_i), \theta_k) x_{ij}$ 
6: end for
7:  $(J_{\mathbf{p}_i}^{\bar{k}}, w_{\bar{k}}) \leftarrow \text{SORT-SCORES}(J_{\mathbf{p}_i}^k, w_k)$ 
8:  $\bar{k}^* = \operatorname{argmax}_{\bar{k} \in \{1, \dots, N\}} \left\{ J_{\mathbf{p}_i}^{\bar{k}} \mid \sum_{i=1}^{\bar{k}} w_i \leq \epsilon_i \right\}$ 
9: return  $(J_{\mathbf{p}_i}^{\bar{k}^*})$ 
```

---

## V. RESULTS AND DISCUSSION

The distributed chance-constrained CBBA algorithm was implemented in simulation and tested on time-critical UAV target tracking missions. The total objective function for each agent is defined as,

$$J_i = \sum_{j=1}^{N_i} \mathbf{R}_{ij}(\tau_{ij}) x_{ij} - f_i d_i(\mathbf{p}_i) \quad (19)$$

where  $d_i(\mathbf{p}_i)$  is the distance traveled by agent  $i$  given path  $\mathbf{p}_i$  and  $f_i$  is the fuel cost per unit distance. The time-critical task rewards are given by,

$$\mathbf{R}_{ij}(\tau_{ij}) = \begin{cases} \mathbf{R}_j e^{-\lambda_j \Delta \tau_{ij}}, & t_{j\text{start}} \leq \tau_{ij} \leq t_{j\text{end}} \\ -\mathbf{R}_j, & \text{otherwise} \end{cases}$$

where reward  $\mathbf{R}_j$  is obtained if the task is done on time, an exponential discount penalizes late tasks according to delay  $\Delta \tau_{ij} = \max\{0, \tau_{ij} - (t_{j\text{start}} + \bar{t}_{j\text{dur}})\}$  (i.e. delay  $\Delta \tau_{ij}$  represents the amount of time in excess of the expected task duration if the task had been started on time), and finally a negative reward of  $-\mathbf{R}_j$  is incurred for failing to do the task within the time-window (e.g. representing loss of resources and opportunity cost associated with committing to a task and failing to perform it). The actual task durations  $t_{j\text{dur}}$  were considered random variables sampled from gamma distributions with mean  $\bar{t}_{j\text{dur}}$ . Three types of tasks were defined: high-reward high-uncertainty tasks, medium-reward tasks with low uncertainty, and low reward tasks but with deterministic service times (same mean duration for all tasks). Two types of teams were considered: (1) homogeneous team of UAVs with uncertain velocities (uniform distribution over speed), where all agents had the same statistical properties; and (2) heterogeneous UAV teams where half the team consisted of fast but unpredictable agents (high mean and high variance), and the other half involved slower speed but more predictable agents (lower mean and lower variance), both having uniform distributions on velocities. The different risk allocation strategies for homogeneous and heterogeneous

agents were compared and the results are described in the following sections.

#### A. Homogeneous Multi-Agent Team

Fig. 3 shows Monte Carlo simulation results comparing chance-constrained mission performance as a function of the mission risk level. In the experiments, the following 7 planning algorithms were compared: a deterministic algorithm (using mean values of parameters), an algorithm optimizing worst-case performance, the chance-constrained CBBA algorithm without explicit risk allocation (all agents planned with mission risk,  $\epsilon_i = \epsilon, \forall i$ , which is typically conservative), chance-constrained CBBA using the different homogeneous risk allocation strategies (Gaussian, Exponential and Gamma), and finally a centralized chance-constrained sequential greedy algorithm (SGA). The chance-constrained mission scores as a function of mission risk are shown in Fig. 3(a) on a linear scale and in Fig. 3(b) on a log-scale to highlight performance at low risk levels. The three risk allocation strategies achieved higher performance than without risk allocation, with the Exponential risk performing best on average. At low risk levels, Gaussian risk allocation gave good performance but as the risk level increased the approximation became worse. All the chance-constrained planning approaches performed significantly better than the deterministic and worst-case planning algorithms that did not account for risk.

Fig. 3(c) shows the achieved team risk corresponding to the given agent risk allocations  $\epsilon_i$ , where the dotted line represents a perfect match between desired and actual mission risk. Without risk allocation the team performs conservatively, achieving much lower mission risk than allowed and thus sacrificing performance. With the risk allocation methods, the team is able to more accurately predict the mission risk, where closer matches led to higher scores. Finally, chance-constrained CBBA achieved performance on par with the centralized sequential greedy approach, validating the distributed approximation to the centralized chance-constrained problem.

#### B. Heterogeneous Multi-Agent Team

Fig. 4 shows Monte Carlo simulation results for a stochastic mission with 6 heterogeneous agents and 60 tasks. In these experiments, the following 8 planning algorithms were compared: deterministic, worst-case, chance-constrained CBBA without risk allocation, chance-constrained CBBA using the risk allocation heuristic proposed in [13] with  $\mathbf{H} = \sqrt{2/N_a}$  (see Eq. (12)), chance-constrained CBBA using the heterogeneous risk allocation strategies proposed in Sec. III (equal shares, shares based on variance, shares based on std. dev.), and the centralized chance-constrained algorithm. As seen in Figs. 4(a)-(b), all the chance-constrained planning approaches did better than the deterministic and worst-case algorithms. The heterogeneous risk allocation strategy proposed in this paper, with shares proportional to std. dev., performed best overall. The heuristic risk allocation of [13] achieved similar performance as well (recall that the two

strategies were shown to be equivalent in Sec. III). The other risk allocation approaches performed rather poorly, even though in the equal share case the achieved team risk matched the desired risk well (see Fig. 4(c)). The intuition behind these results is that when agent risk allocations were severely unequal, some agents developed very aggressive plans whereas others selected plans that were too conservative, without considering the effect on the mission as a whole. As a result, the achieved score distributions were quite different between agents, and the convolved mission score distribution yielded lower chance-constrained scores. In general, having a more equitable risk distribution for the team led to higher performing plans, where approaches that made  $\epsilon_i$  equal for all agents (shares proportional to std. dev.) obtained the highest scores. Once again, chance-constrained CBBA achieved performance on par with the centralized approach, validating the distributed approximation.

## VI. CONCLUSION

This paper proposed new risk allocation strategies for distributed chance-constrained planning in multi-agent multi-task stochastic missions. Building upon previous efforts that extended chance-constrained planning to distributed environments [13], this work presented a more formal approach to risk allocation, and proposed several risk allocation strategies for homogeneous and heterogeneous agents that can be leveraged within the distributed chance-constrained planning framework. In particular, the contributions of this work included: presenting a framework for homogeneous and heterogeneous risk allocation, proposing risk allocation strategies that exploit domain knowledge of agent score distributions to improve team performance, and providing insights and intuition as to what parameters affect these allocations and what features affect the performance of the overall chance-constrained mission score given the distributed approximation. Results demonstrated improved performance in time-critical mission scenarios given allowable risk thresholds.

## ACKNOWLEDGMENTS

This research was supported in part by AFOSR (FA9550-08-1-0086) and MURI (FA9550-08-1-0356). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government.

## REFERENCES

- [1] D. Bertsimas and R. Weismantel, *Optimization over integers*. Dynamic Ideas Belmont, MA, 2005.
- [2] S. S. Ponda, L. B. Johnson, A. Geramifard, and J. P. How, *Handbook of Unmanned Aerial Vehicles*. Springer, 2012 (to appear), ch. Cooperative Mission Planning for Multi-UAV Teams.
- [3] H. Choi, L. Brunet, and J. How, "Consensus-based decentralized auctions for robust task allocation," *Robotics, IEEE Transactions on*, vol. 25, no. 4, pp. 912–926, 2009.
- [4] L. B. Johnson, H.-L. Choi, S. S. Ponda, and J. P. How, "Allowing non-submodular score functions in distributed task allocation," in *IEEE Conference on Decision and Control (CDC)*, Dec 2012 (to appear). [Online]. Available: [http://acl.mit.edu/papers/cdc\\_12.cbba.submitted.pdf](http://acl.mit.edu/papers/cdc_12.cbba.submitted.pdf)

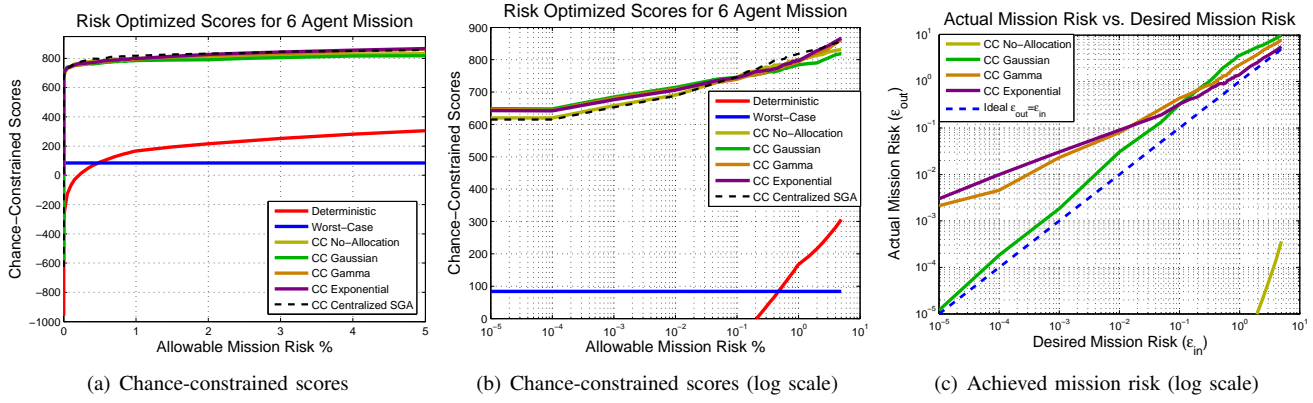


Fig. 3. Monte Carlo results for a stochastic mission with 6 homogeneous agents and 60 tasks, comparing the performance of Chance-Constrained CBBA using different risk allocation strategies.

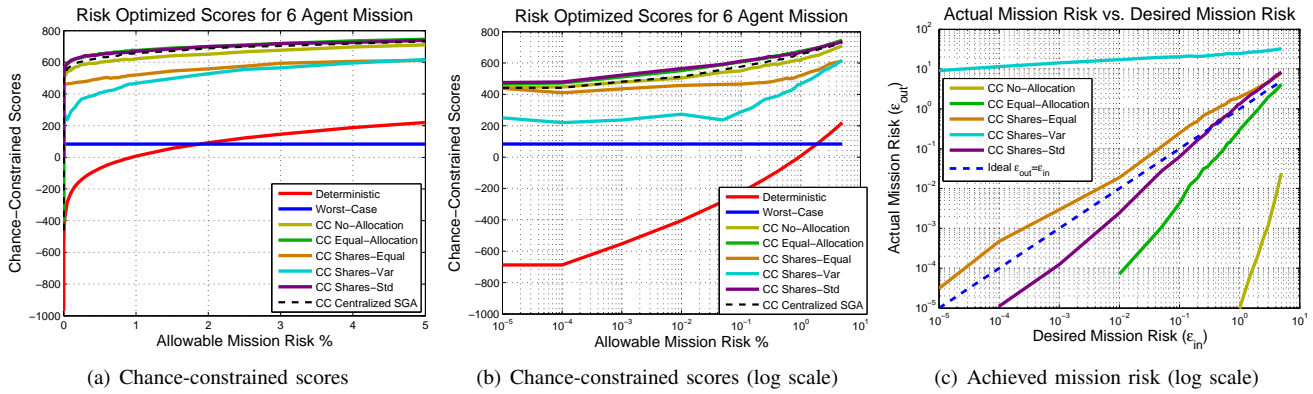


Fig. 4. Monte Carlo results for a stochastic mission with 6 heterogeneous agents and 60 tasks, comparing the performance of Chance-Constrained CBBA using different risk allocation strategies.

[5] S. S. Ponda, "Robust Distributed Planning Strategies for Autonomous Multi-Agent Teams," Ph.D. dissertation, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, September 2012.

[6] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. I-II, 3rd Ed.* Belmont, MA: Athena Scientific, 2007.

[7] D. Bertsimas, D. Brown, and C. Caramanis, "Theory and applications of robust optimization," *SIAM Review*, vol. 53, no. 3, pp. 464–501, 2011.

[8] A. Nemirovski and A. Shapiro, "Convex approximations of chance constrained programs," *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 969–996, 2007.

[9] D. Bertsimas and M. Sim, "The price of robustness," *Operations Research*, vol. 52, no. 1, pp. 35–53, 2004.

[10] A. Ben-Tal and A. Nemirovski, "Robust solutions of uncertain linear programs," *Operations Research Letters*, vol. 25, no. 1, pp. 1–14, 1999.

[11] E. Delage and S. Mannor, "Percentile optimization for markov decision processes with parameter uncertainty," *Operations research*, vol. 58, no. 1, pp. 203–213, 2010.

[12] L. Blackmore and M. Ono, "Convex chance constrained predictive control without sampling," *AIAA Proceedings.[np]. 10-13 Aug*, 2009.

[13] S. S. Ponda, L. B. Johnson, and J. P. How, "Distributed chance-constrained task allocation for autonomous multi-agent teams," in *American Control Conference (ACC)*, June 2012. [Online]. Available: [http://acl.mit.edu/papers/ACC2012-ChanceConstrainedCBBA\\_final\\_submitted.pdf](http://acl.mit.edu/papers/ACC2012-ChanceConstrainedCBBA_final_submitted.pdf)