2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Scenario-based robust scheduling for collaborative human-UAV visual search tasks

Luca F. Bertuccelli *Member, IEEE* and Mary L. Cummings *Senior Member, IEEE*

Abstract— **Many decision support algorithms used to aid human decision making provide guarantees of optimal performance when the optimization parameter are perfectly known. However, algorithm performance degrades when the parameters are inappropriately estimated, and furthermore, algorithm performance is also sensitive to the uncertainty arising from human oversight and interaction with the algorithm. This paper discusses decision support in the form of a scheduling algorithm designed to support human search of imagery collected by unmanned aerial vehicles (UAVs). We demonstrate the sensitivity of the algorithm to uncertainty in human search times, and present a new robust formulation for the search recommendation using data obtained from a previous human-in-the-loop experiment. We show that this robust formulation results in fewer constraint violations in the search task recommendations, as well as increased average performance than an algorithm that does not take this uncertainty into account. For the final draft, the goal is to present human-in-the-loop results that confirm the predictions obtained in the simulations.**

I. Introduction

As complex integrated human-machine systems are fielded in the real world [1], [2], there is an increased interest in understanding the benefit that decision support can provide to the human supervisors. For example, multi-sensor systems Gorgon Stare will push the limits of human performance as intelligence analysts analyze large databases of imagery in time-pressured environments [3], and key questions, such as what constitutes adequate decision support, will need to be answered in order to enable successful operational implementation of these systems.

The integration of human supervision with algorithm support presents an interesting set of challenges. On the one hand, decision support systems are enabled by powerful algorithms that can rapidly solve complex optimizations that are otherwise infeasible for a human operator to consider. Yet, the stochastic dynamics of the environment make it difficult for algorithm designers to precisely specify the parameters in order to guarantee algorithm performance in the presence of parameter uncertainty. Furthermore, human supervision of these systems introduces additional elements of stochasticity that can make algorithm performance sensitive to this interaction. For example, UAV operators will be responsible for making important planning decisions, but the decision times and outcomes are inherently stochastic [4], [5], and this variability becomes an issue for planning systems as the optimization is sensitive to uncertainty in this information [6]–[8].

L. Bertuccelli is with the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 77 Massachusetts Av. 33-407, Cambridge, MA, 02139 USA e-mail: lucab@mit.edu. M. Cummings is with the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 77 Massachusetts Av. 33-311, Cambridge, MA, 02139 USA e-mail: missyc@mit.edu.

With the advent of pervasive sensing in human supervisory control applications, the goal of our work is to design a resilient decision support system that can aid human operators in deciding which visual search tasks must be analyzed, and in which order. Unfortunately, the parameters in this visual search task prioritization problem, such as human search times, are highly uncertain and will generally degrade the performance of the decision aid if the uncertainty is ignored [7]. Our previous work has demonstrated that variability in human search times can lead to performance variations in the scheduling of imagery tasks to be search [5], [7], and work by other authors in other fields has suggested that optimization algorithms can be "brittle" due to the effects of unmodeled uncertainty [9]–[11]. While brittleness can be interpreted as a sensitivity problem to both the structural and parametric modeling assumptions made in the algorithms, the goal of this paper is to address the parametric sensitivity problem of the optimization algorithms, specialized to scheduling a set of search tasks with uncertain processing times.

One possible method for mitigating the effects of uncertainty is to use sensitivity analysis to determine the importance of variations in different parameters. Sensitivity analysis of scheduling problems remains an open problem [12], [13], but it is well understood that uncertainty in start times and processing times can degrade performance of machine shop and job-shop scheduling algorithms [14]–[16]. Sensitivity analysis can also be used to identify parameters that drive the performance degradation of the algorithm, and indirectly enable the provision of a set of alternative schedules to improve performance [17]. Another consideration is the problem of parameter uncertainty from a robust optimization perspective, in which the goal is to generate solutions that are "immune" to the uncertainty. Previous work has presented robust optimization techniques for scheduling under uncertainty [18]–[25] but little work has been presented in applying robust scheduling algorithms to the collaborative human-UAV visual search task problem. For an example of the issue of parameter sensitivity in scheduling multi-UAV missions with operator supervision, see Ref. [26].

This paper tackles the problem of scheduling search tasks for human operators in a UAV domain using a robust optimization approach. In this problem, the human is modeled as a single server that is presented with a stream of randomly arriving tasks which need to be processed, or more specifically, searched. The key idea is to provide a robust decision support algorithm that can help direct the operator's attention based on the priority of the tasks, but also be resilient to variability in operator performance arising from stochastic search times. We show that such a robust decision support has the potential of improving overall human-system performance by reducing the number of constraint violations and improving worst-case performance in the scheduling problem. The results of this work are not restricted to search problems, however, but can also be applied to more general human supervisory settings which do not have deterministic components, such as robust planning for human-supervised power plants.

This paper is outlined as follows: Section II presents the scheduling formulation for the visual search task problem and Section III presents the robust scheduling formulation. We consider numerical results in Section IV and conclude with ongoing and future work.

II. Scheduling Formulation

An interpretation for collaborative human-UAV search tasks is a general queueing model for multi-UAV supervisory control problems involving a visual search task. Search tasks are generated by a Poisson process at an average rate λ, and the human operator (with possible help from a decision support system, DSS) services the tasks at a rate λ_e. In complex tasks, operators may dedicate themselves only to a single task at time, allowing the incoming tasks to accumulate in the queue. In different implementations of the queueing models, operators may be required to make decisions on each task before they move on to another task, or can optionally requeue a task and investigate the image at a later time [4]. The visual search task initiates when the operator begins examining the image feed, and concludes with a decision on the target location.

A. Scheduling objective

We assume that a given time in the queue there are N heterogeneous search tasks known to the operator, where each task $\mathcal{T}_j$ is defined by the following 3-tuple $\mathcal{T}_j = \{R_j, t_j, s_j\}, \quad \forall j = 1, \ldots, N$. In this description, R_j is the reward for succesfully completing task j, t_j is the total time required to complete or service the task, and s_j indicates the time at which the task becomes available to the operator. The operator scheduling problem is defined in this paper as the goal of maximizing the total accumulated reward obtained from a sequence of search tasks in a finite mission time T_H.

A schedule, $\mathcal{S}$, is an ordered list of tasks, consisting of $\mathcal{S} = \{k_1, k_2, \ldots, k_N\}$ where k_i is the index of the task in the i^{th} location in the schedule. The reward of the task is specified clearly to the operator, so that rational decision theory can be applied [27]. The start time s_j for each vehicle is used to model the fact that different tasks may become available at different times, either due to vehicle constraints or because image feeds are unavailable due to communication dropouts. UAVs that are not being used by the operator to search can loiter around their intended targets. In this paper we assume that each UAV is assigned exactly one task, as an abstract representation of real-world surveillance systems such as Gorgon Stare [3].

In the more general case when all task imagery is not available at the same time, $s_j \neq s, \quad \forall j$, the problem can be formulated as a modification of the well-studied single machine scheduling problem with arbitrary release dates [29]. The distinguishing feature from the classical machine scheduling problem is that for our problem formulation, the operator seeks to maximize the accumulated reward of the total mission, which is slightly different from the classical machine scheduling paradigm which seeks to minimize the total delay, total number of tardy jobs, and/or flowtimes [29].

With the goal of maximizing the accumulated reward, the scheduling formulation takes the following form

$$\mathbf{SP} \left\{ \begin{array}{lll} & \max_{x_k^j \in \{0,1\}} \sum_j \sum_k R_j x_k^j & \\ \text{subject to:} & \sum_k x_k^j \leq 1 & \forall j \\ & \sum_j x_k^j \leq 1 & \forall k \\ & \sum_j (s_j + t_j) x_k^j \leq C_k, & \forall k \\ & C_{k-1} + \sum_j t_j x_k^j \leq C_k, & \forall k \\ & C_k \leq T_H, & \forall k \end{array} \right.$$

In problem **SP** the notation x_k^j denotes the position k in which task j appears. Note that with this formulation introduces an additional non-negative variable, C_k which denotes the completion time of the k^{th} task (which cannot exceed the total available time T_H).[1] Two constraints are required for this optimization: first, the k^{th} task cannot be completed prior to the sum of its scheduled time $\sum_j s_j x_k^j$ and the search time $\sum_j t_j x_k^j$ (the third constraint). Second, the next task k cannot be completed before the previous task is completed, and the completion time of the previous task, C_{k-1} (the fourth constraint). Note that this paper does not assume preemption [29], meaning that the a task cannot be interrupted in order to initiate another task.

B. Uncertain processing times

In the case when the processing times are human search times, it is well known in the visual search literature that human search times are context-specific and are not well described with a deterministic representation [31]. In the specific application to the UAV search problem, extensive human-in-the-loop experiments of simulated UAV missions have shown that search times are uncertain [4], [5], [30].

Nonetheless, results from previous experimental data regarding the visual search task in a simulated multi-UAV experiment [4], [5], [30] have shown the randomness in the search times can be characterized with a log-normal distribution in Eq. (1)

$$f(t_s; \bar{T}, \sigma^2) \propto \exp\left(\frac{-(\log(t_s) - \bar{T})^2}{2\sigma^2}\right), \quad t_s > 0 \qquad (1)$$

[1] Note that the schedule and/or the completion times need not be unique, since if there is sufficient time to perform the tasks, it may be possible to either rearrange the task list, as well as delay the start of the task. Delaying the task may be beneficial in a setting when the tasks are arriving randomly.

Here $\bar{T}$ and σ^2 are the maximum likelihood estimates of the mean and variance of the search times that have been obtained from the previous experimental data [4].

C. Encoding uncertain processing times in the schedule

Unfortunately there is no guarantee for the optimality of the schedules determined by **SP** when the processing times t_j are uncertain. For example, a schedule that is generated by assuming known parameters can be infeasible when implemented on a system since the realization of the processing times may vary greatly from the fixed parameters assumed in the deterministic optimization. Infeasibility arises when the schedule exhibits suboptimal performance because not all the search tasks may be processed in the allotted time T_H, and the total mission time constraint is violated. In a human supervisory control setting, an algorithm that outputs an infeasible solution when implemented in a real system can result in loss of trust by the human supervisor [32].

Previous work in stochastic scheduling has made significant progress on the role of uncertainty in the processing times. For example, if a prior distribution on the search times is available, stochastic programming can be used to generate schedules that are optimal on average [33]. While this is often a desirable objective, the stochastic programming approach does not explicitly account for the variance of the solution, and seeks to find a solution such that all constraints are satisfied under all realizations of the uncertain times [23]. In the case when no schedule can satisfy all the constraints, the schedule is deemed infeasible and no recommendation is made.

An alternative to the stochastic programming formulation is to consider reactive schedules, in which a replan is instantiated when new information appears. The key to a successful implementation of reactive planning relies of finding computationally efficient method for generating the schedules. One way to avoid the computational explosion of stochastic programming methods relies on a suboptimal approach that replaces the uncertain processing times with their expected values. By adopting this naïve certainty equivalence (CE) approach, the optimization solves the following objective for equal start times ($s_j = s$)

$$\textbf{CE} \quad \left\{ \max_{x_j \in \{0,1\}} \sum_{j=1}^{N} R_j x_j \mid \sum_j \bar{t}_j x_j \leq T_H \right\} \tag{2}$$

Note that the **CE** approach does not truly account for the variability in the search times, but allows implementations in which a reactive plan is generated quickly.

III. Scenario-based robust scheduling

Another approach that proactively plans with uncertainty in the processing times is a robust optimization approach to the scheduling problem [19]–[24]. Robust optimization is beneficial in that it allows the optimized solution to be somewhat "immune" to the uncertainty. For example, a robust solution will generally have a higher guarantee of performance under a wide range of realizations of the uncertainty, while possibly paying a performance penalty in a nominal setting with little uncertainty.

Many different robust scheduling methods have been proposed in the literature such as β-robust schedules using constraint programming that probabilistically guarantee a desired level of suboptimality [19] or by prescribing infeasibility tolerances and reliability levels [24]. Other techniques solve a scenario-based optimization [22], [23], and we pursue this latter option since scenarios are easy to generate from distributions obtained from our previous experiments and retain the integer programming formulation of the original problem, whereas alternative formulations might increase the complexity of the original problems [34].

A. Homogeneous start times

For the case when all the tasks are available at the same time, the robust formulation with uncertain processing times takes the form of

$$\min_{\tilde{t} \in \mathcal{T}} \max_{x_j \in \{0,1\}} \sum_{j=1}^{N} R_j x_j \mid \sum_j \tilde{t}_j x_j \leq T_H \tag{3}$$

In this setting, the maximization is performed with respect to the decision variable x, but "nature" aims to minimize the objective, over an uncertainty set $\mathcal{T}$ on the processing times $\tilde{t}$. For the case of a distribution with semi-infinite support, such as the log-normal distribution, this optimization might be overly conservative, and thus is desirable to use scenarios to generate a solution that hedges against possible adverse realizations of processing times [23]. Scenario-based robust implementations of the optimization in Eq. (3) for equal start times can be formulated as follows [22]

$$\textbf{RKP} \quad \left\{ \begin{array}{ll} & \max_{x_j \in \{0,1\}} \sum_{j=1}^{N} R_j x_j - \beta \gamma^+ \\ \text{subj. to:} & \sum_j \tilde{t}_j x_j - T_H \leq \gamma \\ & \gamma \geq 0 \end{array} \right.$$

where $\gamma^+ \doteq \max(0, \gamma)$. In this formulation, the key idea is to maximize the reward but penalize instances when the total mission time exceeds the mission time T_H. Note that Eq. (4) only penalizes mission *delays*, when the total mission time exceeds T_H. If the task schedule does not exceed the time T_H, then the penalty is identically zero. The tuning parameter β reflects designer aversion to constraint violations, and in the case when $\beta = 0$ recovers the case when the designer is insensitive to schedules which could exceed the allotted mission time.

Implementation of **RKP** requires the use of scenarios for the processing times, where $\tilde{t}_j \sim f(t_s; \bar{T}, \sigma^2)$ from Eq. (1). For the case where each task search time is independent and identically distributed, then the scenario-based robust knapsack problem **SB-RKP** can be formulated as follows

$$\textbf{SB-RKP} \left\{ \begin{array}{ll} & \max_{x_j \in \{0,1\}} \sum_{j=1}^{N} R_j x_j - \beta \gamma \\ \text{subj. to:} & \sum_j \tilde{t}_j^1 x_j - T_H \leq \gamma \\ & \sum_j \tilde{t}_j^2 x_j - T_H \leq \gamma \\ & \vdots \\ & \sum_j \tilde{t}_j^{N_s} x_j - T_H \leq \gamma \\ & \gamma \geq 0 \end{array} \right.$$

In this optimization, each $\tilde{t}_j^m$ is the m^{th} scenario of the search time for the j^{th} task, and there are a total of N_s scenarios. Note that the penalty term γ^+ of Eq. (4) is implemented by enforcing the constraint that $\gamma \geq 0$, and augmenting the cost function with the term $-\beta\gamma$. The number of constraints is now linear in the number of scenarios, $O(N_s)$.

B. *Heterogeneous start times*

For the case of heterogeneous start times, when $s_j \neq s$, the robust optimization is a modification of **SP**. The scenario-based robust scheduling problem **SB-RSP** can be formulated as follows

$$\textbf{SB-RSP} \left\{ \begin{array}{ll} & \max_{x_k^j \in \{0,1\}} \sum_j \sum_k R_j x_k^j - \beta\gamma \\ \text{subj. to:} & \sum_k x_k^j \leq 1 \quad \forall j \\ & \sum_j x_k^j \leq 1 \quad \forall k \\ & \sum_j (s_j + t_j^m) x_k^j \leq C_k^m, \quad \forall k, m \\ & C_{k-1}^m + \sum_j t_j^m x_k^j \leq C_k^m, \quad \forall k, m \\ & C_k^m - T_H \leq \gamma, \quad \forall k \\ & \gamma \geq 0 \end{array} \right.$$

Similar to the **SB-RKP** problem, the major differences from **SP** are in additional constraints that are added for the scenarios. Effectively, there is now a different completion time C_k^m for each of the m scenarios of the search times t_j^m. The third and fourth constraints define the completion times for the k^{th} task in the optimal schedule, the fifth constraint ensures that the completion times are less than the mission time T_H by a term γ. If all the scenarios used in **SB-RSP** result in a schedule that can be completed in time T_H, then $\gamma = 0$, and there is no penalty term included in the objective function. Note that the number of constraints in this optimization is now on the order of $O(N^2 + NN_s)$, which is a increase in the number of constraints that is linear in the number of scenarios.

In summary, we have presented two new scenario-based formulations for the visual search task optimization problem in the presence of operator search time uncertainties. We evaluate the role of the tuning parameter β in the simulations shown in the next sections.

IV. Evaluation of Robust Solutions

This section presents numerical experiments showing the benefit of the scenario-based robust solutions over the certainty equivalent methods that could be used in visual search tasks. All optimizations have been implemented in GLPK, but it is straightforward to implement them in state of the art commercial solvers such as CPLEX. In the first section, we consider the homogeneous start time cases and the heterogeneous start times will be considered next.

A. *Homogeneous start times*

For the first simulations, the optimization **SB-RKP** is solved with $N_s = 100$ scenarios and we varied the time horizon in the range $T_H = [50 : 50 : 500]$. The units of T_H are time, but can be considered non-dimensional for the purposes of these simulations. For each time horizon T_H, we solved the corresponding nominal optimization **CE** and robust solution **SB-RKP** and found the optimal actions for each optimization.

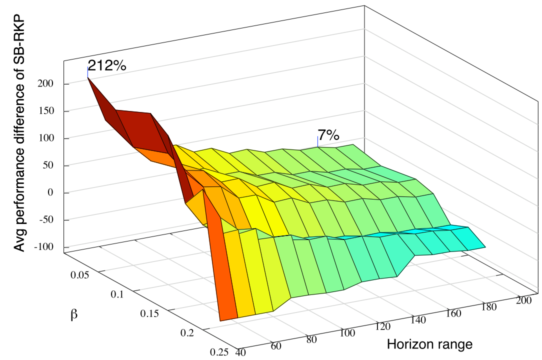

Fig. 1. Average performance difference of **SB-RKP** compared to **CE**, as a function of β and different time horizon

Figure 1 shows the range of values for which the choice of β (on the x-axis) and time horizon T_H (y-axis) impact the performance of the robust solution **SB-RKP** compared to the nominal **CE** performance. In these numerical simulations the goal is to evaluate the different in *mean* performance of the robust actions compared to the nominal actions. To calculate the mean performance, the robust and nominal objectives were calculated with realizations of the start times, and the total reward that was accumulated prior to exceeding the mission time T_H was averaged for the robust and nominal actions respectively. Figure 1 shows that a moderate level of conservatism can significantly improve the **CE** performance by a factor of 2 for shorter time horizons, as expected due to the high number of constraint violations for shorter time horizons. Even for longer time horizons, as the number of constraint violations of the **CE** optimization decrease, the robust optimization can still present improvements on the order of 7%. Note that being overly conservative ($\beta \approx 0.2$) can however also result in worst performance on average, and thus users must be careful to tune the optimization parameter appropriately.

B. *Heterogeneous search times*

For the heterogeneous search times, our numerical results look at the variability of the robust solution of **SB-RSP** to the number of scenarios but also present some characteristics of the robust solution that provide intuitive appeal to human supervision. Scenario-based optimizations are highly dependent on the total number of scenarios, though the number of required scenarios to achieve robust performance is highly dependent on the application context and remains an open problem [22]. Figure 2 shows the quality of the robust solution for $N = 20$ tasks as a function of the total number of scenarios used in the optimization. As expected, since the goal of the optimization is to maximize the reward, the robust solution is expected to decrease (becoming more

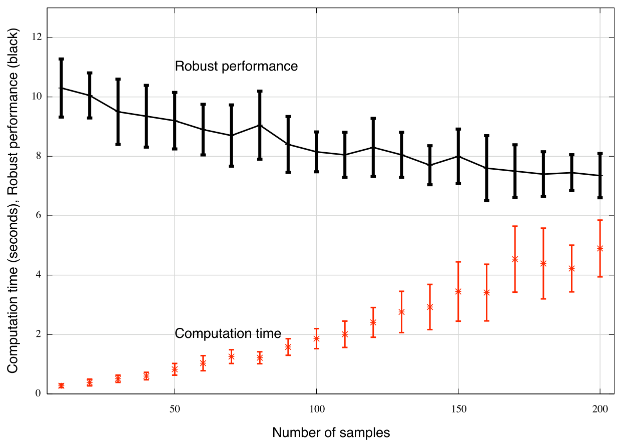

Fig. 2. Effect of samples on the computation time and robust performance. Note that as the number of scenarios increases, the computation time for this small problem scales almost linearly, even though the scaling will be exponential due to the addition of a linear number of constraints.

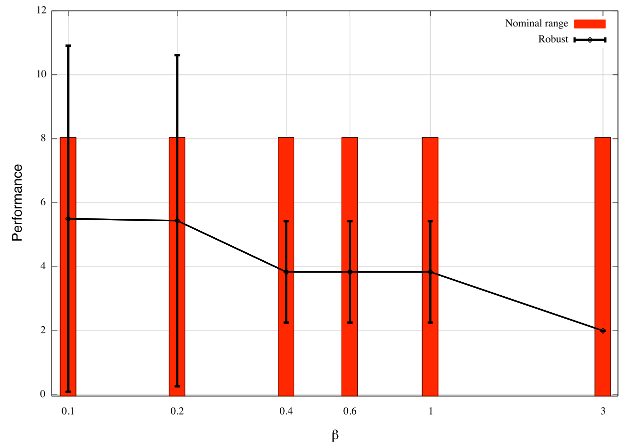

Fig. 3. Sensitivity of the robust solution to higher robustness achieved by increasing β. The robust solution changes in mean performance from 5.44 to 3.84, but the standard deviation improves from 2.70 (for $\beta = 0.1$) to 0.79 (for $\beta = 1$).

conservative) as the number of scenarios is increased. Note that the computation times should be used for trend information only, since the GLPK implementation will necessarily be much slower than with a commercial solver such as CPLEX. While the number of scenarios presents a linear increase in the total number of constraints, the scheduling formulation is known to be NP-hard, and therefore the computation time is expected to increase in an exponential manner.

C. Robust solution properties

1) Robust performance: A key benefit of the robust optimization is to improve on the variability of the nominal solution. Table I shows the results from 1000 Monte Carlo simulations for a scenario where $R \in \mathcal{U}(5, 10)$, while for the log-normal distribution $\mu = 10$, $\sigma^2 = 1 + i/10$, $\forall i = 1, 2, \ldots, N$. While the **CE** results in a slightly higher average performance of 16.99 compared to the **SB-RSP** solution of 14.99, the **CE** solution has a higher standard deviation of $\sigma = 2.25$, compared to $\sigma = 0.12$ of the robust solution. This implies that in a real mission, the overall system could achieve a much lower reward than that predicted by the **CE** optimization, and could lead to trust and acceptance issues by the human operator.

Figure 3 demonstrates the sensitivity of the robust solution to the choice of the tuning parameter β. The red bars show the 2-σ range of the **CE** solution, which is of course invariant to the choice of β, but is included for visualization purposes. The performance of the robust solution is shown in the black line, with 2-σ range. A higher level of robustness is achieved by increasing β: note that as the robustness increases, the average performance decreases, but comes at the benefit of a reduction in the variance of the robust solution.

2) Mission times: A key attribute in understanding the robust solution is to understand how well the schedules generated by the **SB-RSP** satisfy the temporal constraint of achieving the maximum reward of tasks within a time T_H. Figure 4 shows a histogram of the difference between

TABLE I

PERFORMANCE DIFFERENCE BETWEEN **CE** AND **SB-RSP**

	Mean	Standard deviation
CE	16.99	2.25
SB-RSP	14.99	0.12

the mission times of the robust optimization of **SB-RSP** (in black) compared to those of the nominal optimization **CE** (in red). A negative difference means that the schedule has retained some margin in the mission time, while a positive difference indicates that the schedule has exceed the mission time constraint. The **SB-RSP** histogram is centered around -15, while the **CE** is centered around -5, suggesting that on average, the **SB-RSP** provides increased buffer to satisfying the mission time constraint. Note that the robust optimization creates schedules whose tasks will exceed the total mission time with very low probability, while for these simulations, the **CE** optimization exceeds the mission time 16.8% of the time. This set of constraint violations in turn results in the suboptimal performance observed in Table I. In summary, the robust optimization hedges against placing too many tasks in a bundle to present to the operator by recommending tasks that can be completed with high probability, given that they are uncertain. The downside is that there is a price to be paid in terms of nominal performance, but this can be tuned by the designer by selecting the parameter β to be aligned with a desired level of conservatism.

V. CONCLUSION

This paper has presented the optimization problem of scheduling search tasks for human operators in a UAV domain using a robust optimization approach. In this problem, the human is modeled as a single server that is presented with a stream of randomly arriving tasks which need to be searched. We have presented a novel scenario-based optimization specialized to the collaborative human-UAV

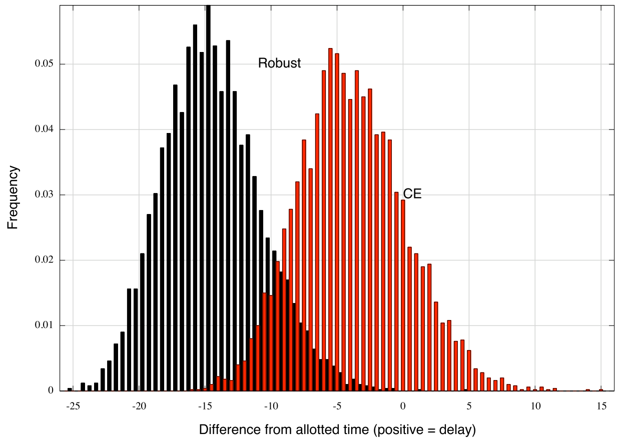

Fig. 4. Difference in completion time showing that a nominal **CE** approach can result in infeasible mission

search and shown that a simple robust optimization technique can significantly enhance the performance of a decision support system, and result in solutions that result in improved performance in the case of uncertain search times.

An important ongoing objective is the understanding of whether robust optimization can also be beneficial in the case of reactive policies, where rather than optimizing around a nominal model, the optimization is performed robustly. This robust adaptive replanning has been beneficial in other optimization settings [35], and it will be important to verify the conditions under which this benefit can be harnessed in the scheduling problem as well.

VI. Acknowledgements

This work was supported by the Michigan/AFRL Collaborative Center in Control Science and under ONR Grant N00014-07-1-0230. Funding gratefully acknowledged from ONR Codes 31 and 34. The authors gratefully acknowledge the comments of Niek Beckers in the manuscript.

References

[1] J. McCarley and C. Wickens, "Human factors implications of UAVs in the national airspace," *Aviation Human Factors Division, Savoy, Illinois, Technical report AHFD-05-05/FAA-05-01*, 2005.

[2] M. Cummings, S. Bruni, and P. Mitchell, "Human Supervisory Control Challenges in Network-Centric Operations," *Reviews of Human Factors and Ergonomics*, vol. 6, no. 1, pp. 34–78, 2010.

[3] M. Hoffman, "New Reaper sensors offer a bigger picture." `http://www.airforcetimes.com/news/2009/02/airforce_WAAS_021609/`.

[4] L. Bertuccelli, N. Pellegrino, and M. Cummings, "Choice Modeling of Relooks Tasks in UAV Search Missions," *IEEE American Control Conference*, 2010.

[5] D. Southern, "Human-guided management of collaborating unmanned vehicles in degraded communication environments," 2010.

[6] M. Cummings, A. Clare, and C. Hart, "The role of human-automation consensus in multiple unmanned vehicle scheduling," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 52, no. 1, p. 17, 2010.

[7] L. Bertuccelli, N. Beckers, and M. Cummings, "Developing Operator Models for UAV Search Scheduling," in *AIAA Guidance Navigation and Control Conference*, Aug 2010. AIAA-2010-7863.

[8] L. Bertuccelli, M. Alighanbari, and J. How, "Robust planning for coupled cooperative UAV missions," in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 3, pp. 2917–2922, IEEE, 2004.

[9] D. Aha, "Tolerating noisy, irrelevant and novel attributes in instance-based learning algorithms," *International Journal of Man-Machine Studies*, vol. 36, no. 2, pp. 267–287, 1992.

[10] B. Silverman, "Human-computer collaboration," *Human–Computer Interaction*, vol. 7, no. 2, pp. 165–196, 1992.

[11] P. Smith, C. McCoy, and C. Layton, "Brittleness in the design of cooperative problem-solving systems: The effects on user performance," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 27, no. 3, pp. 360–371, 2002.

[12] B. Penz, C. Rapine, and D. Trystram, "Sensitivity analysis of scheduling algorithms," *European Journal of Operational Research*, vol. 134, no. 3, pp. 606 – 615, 2001.

[13] N. Hall and M. Posner, "Sensitivity analysis for scheduling problems," *Journal of Scheduling*, vol. 7, no. 1, pp. 49–83, 2004.

[14] B. Penz, C. Rapine, and D. Trystram, "Sensitivity analysis of scheduling algorithms," *European Journal of Operational Research*, vol. 134, no. 3, pp. 606–615, 2001.

[15] N. Hall and M. Posner, "Sensitivity analysis for scheduling problems," *Journal of Scheduling*, vol. 7, no. 1, pp. 49–83, 2004.

[16] Z. Li and M. Ierapetritou, "Process scheduling under uncertainty: Review and challenges," *Computers & Chemical Engineering*, vol. 32, no. 4-5, pp. 715–727, 2008.

[17] Z. Jia and M. Ierapetritou, "Short-term scheduling under uncertainty using MILP sensitivity analysis," *Ind. Eng. Chem. Res*, vol. 43, no. 14, pp. 3782–3791, 2004.

[18] W. Herroelen and R. Leus, "Project scheduling under uncertainty: Survey and research potentials," *European journal of operational research*, vol. 165, no. 2, pp. 289–306, 2005.

[19] C. Wu, K. Brown, and J. Beck, "Scheduling with uncertain durations: Modeling β-robust scheduling with constraints," *Computers & Operations Research*, vol. 36, no. 8, pp. 2348–2356, 2009.

[20] R. Daniels and J. Carrillo, "β-Robust scheduling for single-machine systems with uncertain processing times," *IIE transactions*, vol. 29, no. 11, pp. 977–985, 1997.

[21] R. Daniels and P. Kouvelis, "Robust scheduling to hedge against processing time uncertainty in single-stage production," *Management Science*, vol. 41, no. 2, pp. 363–376, 1995.

[22] J. Mulvey, R. Vanderbei, and S. Zenios, "Robust optimization of large-scale systems," *Operations research*, vol. 43, no. 2, pp. 264–281, 1995.

[23] M. Laguna, "Applying robust optimization to capacity expansion of one location in telecommunications with demand uncertainty," *Management Science*, vol. 44, no. 11, pp. 101–110, 1998.

[24] S. Janak, X. Lin, and C. Floudas, "A new robust optimization approach for scheduling under uncertainty: II. Uncertainty with known probability distribution," *Computers & chemical engineering*, vol. 31, no. 3, pp. 171–195, 2007.

[25] X. Lin, S. Janak, and C. Floudas, "A new robust optimization approach for scheduling under uncertainty: I. Bounded uncertainty," *Computers & chemical engineering*, vol. 28, no. 6-7, pp. 1069–1085, 2004.

[26] J. Crandall and M. Cummings, "Identifying predictive metrics for supervisory control of multiple robots," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 942–951, 2007.

[27] J. Berk, E. Hughson, and K. Vandezande, "The price is right, but are the bids? An investigation of rational decision theory," *The American Economic Review*, vol. 86, no. 4, pp. 954–970, 1996.

[28] D. Bertsimas, *Introduction to Linear Optimization*. Athena Scientific, 1997.

[29] K. Baker and D. Trietsch, *Principles of Sequencing and Scheduling*. Wiley-Blackwell, 2009.

[30] C. Nehme, *Modeling Human Supervisory Control In Heterogeneous Unmanned Vehicle Systems*. PhD thesis, MIT, 2008.

[31] W. Schneider and R. Shiffrin, "Controlled and automatic human information processing: I. Detection, search, and attention," *Psychological Review*, vol. 84, no. 1, pp. 1–66, 1977.

[32] J. Lee and K. See, "Trust in automation: Designing for appropriate reliance," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 46, no. 1, p. 50, 2004.

[33] J. Birge and M. Dempstert, "Stochastic programming approaches to stochastic scheduling," *Journal of Global Optimization*, vol. 9, no. 3, pp. 417–451, 1996.

[34] D. Bertsimas and M. Sim, "The price of robustness," *Operations research*, pp. 35–53, 2004.

[35] L. Bertuccelli, B. Bethke, and J. How, "Robust adaptive markov decision processes in multi-vehicle applications," in *American Control Conference, 2009. ACC'09.*, pp. 1304–1309, IEEE, 2009.