2020 American Control Conference
Denver, CO, USA, July 1-3, 2020

Optimal Real-time Scheduling of Human Attention for a Human and Multi-robot Collaboration System

Ningshi Yao Fumin Zhang

Abstract— **We analyze a human and multi-robot collaboration system and propose a method to optimally schedule the human attention when a human operator receives collaboration requests from multiple robots at the same time. We formulate the human attention scheduling problem as a binary optimization problem which aims to maximize the overall performance among all the robots, under the constraint that a human has limited attention capacity. We first present the optimal schedule for the human to determine when to collaborate with a robot if there is no contention occurring among robots' collaboration requests. For the moments when contentions occur, we present a contention-resolving Model Predictive Control (MPC) method to dynamically schedule the human attention and determine which robot the human should collaborate with first. The optimal schedule can then be determined using a sampling based approach. The effectiveness of the proposed method is validated through simulation that shows improvements.**

I. INTRODUCTION

Recent advances in robotics have enabled the reduction in price, size, and operational complexity of robots. A natural outgrowth of these advances are systems comprised of large numbers of robots that collaborate autonomously in diverse applications. However, even though the autonomous task execution capabilities of robots have progressed rapidly, the human's advantage in high-level reasoning and planning is still needed. As a consequence, the form of human and multi-robot collaboration systems has become a popular and important topic [1], [2]. For human and robots collaboration systems, as the human labor cost increases, it can be envisioned that the number of robots that one human needs to work with will increase to a large extent. However, a human has limitation on attention capacity. In psychology studies [3]–[5], researchers discovered that a human can pay attention to only two to four items at the same time. Therefore, when a human is collaborating with multiple robots, the human operator cannot effectively serve or collaborate with all robots at the same time. Which robot the human operator should collaborate with first is a general question for human and multi-robot collaboration systems.

How to allocate or schedule a human's attention to each robot is a research topic studied in real-time scheduling. Well-known scheduling policies, such as Rate Monotonic Scheduling (RMS), Earliest Deadline First (EDF) and First Come First Serve (FCFS) [6] scheduling algorithms, are widely used in real-time systems. These algorithms are optimal in the sense that they can maximize the number of tasks that can be scheduled before their deadlines. However, they are not designed to achieve an optimal control performance. Inappropriately scheduling a human operator to collaborate with robots has been found to have a negative effect on overall performance in human-robot systems [7]. Numerous research efforts have explored how to better schedule a human's attention to robots. In [8], the authors compared two types of scheduling methods, Open-Queue (OP) and Shortest Job First (SJF) scheduling, and showed that SJF scheduling can provide more stable robot performance. In [9], the authors proposed a Highest Trust First (HTS) scheduling based on a robot performance model from [10] and a human–robot mutual trust model, to determine the human operator's schedule to interact with one robot at each time such that the human–robot trust level can always be maintained within a proper range. However, both the SJF and HTS cannot guarantee that the overall performance of robots can be optimized. Murray et al. formulated an integer programming problem to effectively schedule multiple unmanned aerial vehicles and humans to time-sensitive geographically-dispersed tasks and optimize the overall system performance [11]. The integer optimization problem was solved by the IBM CPLEX solver. Although this method can ensure optimiality, a major disadvantage is the computation requirement.

Our earlier work established a contention-resolving Model Predictive Control (MPC) framework for co-designing scheduling and control law for systems with constrained resource [12], [13]. For a human and multi-robot collaboration system, the limited human attention is viewed as the constrained resource. In this paper, we present a contention-resolving MPC design to find the optimal schedule for human attention. We assume that the collaboration between the human and a robot is non-preemptive, which is different from the work [9] where the collaboration between the human operator and a robot can be interrupted. This constraint gives the advantage to reduce the collaboration switches for the human operator and avoid human delays [14]. The contributions of this paper are as follows: (i) We rigorously show that for the case where no contention occurs among robots, the optimal schedule for a robot to maximize its performance is to start the collaboration with a human operator once the collaboration request is generated. For the case where contentions occur, the optimal schedule for a robot i is to start the collaboration right after the time instant when all contended robots that are scheduled to collaborate

Yao and Zhang were partially supported by ONR grants N00014-19-1-2556, N00014-19-1-2266 and N00014-16-1-2667; NSF grants OCE-1559475, CNS-1828678, and S&AS-1849228; NRL grants N00173-17-1-G001 and N00173-19-P-1412 ; and NOAA grant NA16NOS0120028.

Ningshi Yao and Fumin Zhang are with the School of Electrical and Computer Engineering, Georgia Institute of Techonology, Atlanta, GA 30308, USA. Email: {nyao6, fumin}@gatech.edu

before robot i complete their collaborations. This property, which we call *Condition of Immediate Access* or *CIA*, ensures that the human attention scheduling problem satisfies the fundamental assumptions in a real-time scheduling theory. (ii) Based on the non-preemptive collaboration setup, a new analytical timing model is established to determine the significant moments during the human attention scheduling process and accurately compute the time delays caused by contentions given a specific schedule of which robot gets to collaborate with human first. (iii) We formulate an integer optimization problem to design an optimal schedule of human and robot collaboration when contentions occur, in order to maximize the overall performances of all robots. We propose a discrete-time contention-resolving MPC method to solve the integer optimization problem. And we show that the delays introduced by contentions will increase cost in the optimization problem, which guarantees that contention-resolving MPC can find the optimal solution. The effectiveness of our method is verified through simulations and compared with the HTF scheduling strategy.

II. Problem Formulation

Consider one human operator collaborating with N robots. The human is the expert so if the human is collaborating with a robot, then the human can help the robot to improve its performance. For a robot i where $i=1,...,N$, we first introduce a dynamic model describing its performance as

$$\begin{aligned} P_i(k) = &(1-u_i(k))\,[(1-k_{i,R})P_i(k-1)+k_{i,R}P_{i,\min}] \\ &+u_i(k)\,[(1-k_{i,H})P_i(k-1)+k_{i,H}P_{i,\max}] \end{aligned} \tag{1}$$

where k denotes the discrete time step and i denotes the index of a robot. The parameters $P_{i,\min}$ and $P_{i,\max}$ are the minimal and maximal values of the performance value of robot i. The control variable $u_i(k)$ only has two values, 0 or 1. If $u_i(k)=1$, then the robot is in collaborative mode with the human operator. If $u_i(k)=0$, then the robot is in autonomous mode without the collaboration with the human operator. The parameters $k_{i,R}$ and $k_{i,H}$ are coefficients for autonomous and collaborative mode, respectively, satisfying $0<k_{i,H}<k_{i,R}<1$. The model (1) guarantees that $P_i(k)$ is bounded between $[P_{i,\min},P_{i,\max}]$, given that the initial performance value $P_i(k_0)$ is within $[P_{i,\min},P_{i,\max}]$. The performance value $P_i(k)$ will decrease under the autonomous mode because it is a convex combination of $P_i(k-1)$ and $P_{i,\min}$. And $P_i(k)$ will increase under the collaborative mode because it is a convex combination of $P_i(k-1)$ and $P_{i,\max}$.

Based on the performance model, we introduce the human-to-robot trust. We utilize the human-to-robot trust model in [9] to quantify how good the collaboration experience is for the human operator. The trust is modeled as

$$\mathcal{T}_i(k) = A_i\mathcal{T}_i(k-1) + B_iP_i(k) - C_iP_i(k-1) \tag{2}$$

where the function $\mathcal{T}_i(k)$ represents the trust level from the human operator to robot i at time k. It is determined by the previous trust level $\mathcal{T}_i(k-1)$, the robot performance measures $P_i(k)$ and $P_i(k{-}1)$. The parameters A_i, B_i and C_i are constant coefficients whose values depend on the human operator, robot i and the corresponding collaborative task. The trust level should be within a proper range, i.e.

$$\mathcal{T}_{i,\min} \le \mathcal{T}_i(k) \le \mathcal{T}_{i,\max} \text{ for all } k\in[k_0,k_f] \tag{3}$$

where $\mathcal{T}_{i,\min}>0$ and $\mathcal{T}_{i,\max}>0$ are the lower and upper bounds of the trust level for robot i, respectively and the times k_0 and k_f are the starting and ending time of the scheduling time horizon.

For all N robots, each one needs to execute a sequence of tasks $\Gamma_i=\{\tau_{i,1},\tau_{i,2},...,\tau_{i,n_i},...\}$ where i is the index of a robot and n_i is the task index of robot i. We assume the tasks are all periodic and use the notation T_i to denote the period. Let $\alpha_i(n_i)$ denote the time when robot i starts the collaboration request of n_ith task. For any index n_i, $C_i(n_i)$ is the collaboration time that robot i requires to collaborate with the human operator within the time window $[\alpha_i(n_i),\alpha_i(n_i){+}T_i)$ satisfying $1\le C_i(n_i)<T_i$ for all i and n_i. And at each time $\alpha_i(n_i)$, the performance value $P_i(\alpha_i(n_i))$ of robot i is reinitialized to be $P_i^0(n_i)\in[P_{i,\min},P_{i,\max}]$, because each task in task sequence Γ_i may be different from each other. A collaboration completion time $\gamma_i(n_i)$ is the time step when robot i finishes collaborating with the human operator. Since the system is modeled in discrete time, parameters α_i, C_i, T_i and γ_i are all integers.

Remark 1: In problem setup, it is not required that the human and robot collaboration needs to start at the moment $\alpha_i(n_i)$, but we will show in Section II-B that the collaboration starting at $\alpha_i(n_i)$ is optimal to maximize robot performance value if the human attention limitation is ignored is relaxed.

Assumption 1: For each task τ_{i,n_i}, once the collaboration starts between the human and robot i at time k, it will only ends at time $k+C_i(n_i)$.

This assumption indicates that the collaboration between the human operator and a robot is non-preemptive.

A contention time is defined to be a time when two or more robots request to collaborate with the human operator at the same time. At a contention time, due to the human attention limitation, we make the following assumption

Assumption 2: At any given time, at most one robot can be in collaborative mode with the human operator and all the other robots are in autonomous mode, i.e.,

$$\sum_{i=1}^{N} u_i(k) \le 1 \text{ for all } k. \tag{4}$$

Because of contentions, we introduce the delay variable $\delta_i(n_i)\ge 0$ so that $\gamma_i(n_i)=\alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)$. The time delay variables $\delta_i(n_i)$ depend on $C_i(n_i)$, T_i and $u_i(k)$. In Section III, we will present a timing model which can accurately compute $\delta_i(n_i)$ given $u_i(k)$ for all i and k.

A. Formulation of Model Predictive Control

We formulate a human attention allocation problem to compute optimal scheduling $\mathbf{u}^*(k)=(u_1^*(k),...,u_N^*(k))$ on a time interval $[k_0,k_f]$. Given initial human-robot trust level $(\mathcal{T}_1(k_0),...,\mathcal{T}_i(k_0),...,\mathcal{T}_N(k_0))$ and initial robot performance value $(P_1(k_0), ..., P_i(k_0), ..., P_N(k_0))$ for all i, the optimal scheduling problem is to find values for the optimal $\mathbf{u}^*(k)$ by solving the optimization problem

$$\min_{\mathbf{u}(k)} \sum_{i=1}^{N} \sum_{k=k_0}^{k_f} [P_{i,\max} - P_i(k)] \text{ subject to } (1),(2),(3),(4)$$
$$u_i(k)=0, k\in[\alpha_i(n_i), \alpha_i(n_i)+\delta_i(n_i)(\mathbf{u}(k)) - 1],$$
$$u_i(k)=1, k\in[\alpha_i(n_i)+\delta_i(n_i)(\mathbf{u}(k)), \gamma_i(n_i)(\mathbf{u}(k))] \text{ and}$$
$$u_i(k)=0, k\in[\gamma_i(n_i)(\mathbf{u}(k))+1, \alpha_i(n_i+1)-1]$$
$$\text{for all } n_i \text{ such that } k_0\le\alpha_i(n_i) \text{ and } \alpha_i(n_i+1)\le k_f \quad (5)$$

where $\delta_i(n_i)(\mathbf{u}(k))$ and $\gamma_i(n_i)(\mathbf{u}(k))$ represent that these time instants are implicit functions of $\mathbf{u}(k)$. The cost function aims to increase the robot performance as much as possible to reach the upper bounds. Equations (1) and (2) are system dynamic equations. Constraint (3) aims to maintain the trust level within the range and (4) is the contention constraint where $u_i(k)$'s are coupled. Since $\mathbf{u}(k)$ is a vector of binary integers, the problem is binary optimization problem which is non-convex and difficult to solve.

B. Optimal Solution Without Considering Contention

We will first relax the trust level constraint (3) and the human attention limitation constraint (4) in the problem formulation (5) to find the optimal solution $\mathbf{u}(k)$ to maximize the overall robot performance value among the time horizon $[k_0, k_f]$. After relaxing the two constraints, the problem (5) can be decoupled and is equivalent to

$$\sum_{i=1}^{N}\sum_{n_i=1}^{N_i} \max_{\delta_i(n_i)} \sum_{k=\alpha_i(n_i)}^{\alpha_i(n_i+1)-1} P_i(k) \text{ subject to (1) with } P_i^0(n_i) \text{ and}$$
$$u_i(k)=\begin{cases}0, k\in[\alpha_i(n_i),\alpha_i(n_i)+\delta_i(n_i)-1]\\ 1, k\in[\alpha_i(n_i)+\delta_i(n_i),\alpha_i(n_i)+\delta_i(n_i)+C_i-1]\\ 0, k\in[\alpha_i(n_i)+\delta_i(n_i)+C_i,\alpha_i(n_i+1)-1]\end{cases} \quad (6)$$

where N_i is the largest index of tasks satisfying $\alpha_i(N_i)<k_f$.

Theorem 1: (*CIA condition*) The optimal solution for problem (6) is $\delta_i(n_i) = 0$ for all $1\le n_i\le N_i$.

Proof. We first define the cost for robot i within the time window $[\alpha_i(n_i), \alpha_i(n_i+1)-1]$ to be $J_{i,n_i}(\delta_i(n_i)) = \sum_{k=\alpha_i(n_i)}^{\alpha_i(n_i+1)-1} P_i(k)$. Then we will show that the derivative of $J_{i,n_i}(\delta_i(n_i))$ is less than 0, so $J_{i,n_i}(\delta_i(n_i))$ is strictly decreasing as $\delta_i(n_i)$ increases. For simplification, we will use P_i^0 to represent $P_i^0(n_i)$ in the rest of this proof.

During the time $k\in[\alpha_i(n_i), \alpha_i(n_i)+\delta_i(n_i)-1]$, we have $u_i(k) = 0$. The dynamic of robot i's performance value is $P_i(k) = (1-k_{i,R})P_i(k-1) + k_{i,R}P_{i,\min}$ when $u_i(k) = 0$ according to (1). Then for any $k\in[\alpha_i(n_i), \alpha_i(n_i)+\delta_i(n_i)]$, $P_i(k) = (1-k_{i,R})^{k-\alpha_i(n_i)}P_i^0 + k_{i,R}P_{i,\min}\sum_{\kappa=\alpha_i(n_i)}^{k-1}(1-k_{i,R})^{k-1-\kappa} = (1-k_{i,R})^{k-\alpha_i(n_i)}\left(P_i^0-P_{i,\min}\right)+P_{i,\min}$. The sum of costs among time $[\alpha_i(n_i), \alpha_i(n_i)+\delta_i(n_i)]$ is $J^1_{i,n_i}(\delta_i(n_i)) = \sum_{k=\alpha_i(n_i)}^{\alpha_i(n_i)+\delta_i(n_i)} P_i(k) = P_{i,\min}[\delta_i(n_i)+1] + (P_i^0-P_{i,\min})\dfrac{1-(1-k_{i,R})^{\delta_i(n_i)+1}}{k_{i,R}}$.

Let t_1 denote the time step $\alpha_i(n_i)+\delta_i(n_i)$ and P_i^1 denote $P_i(t_1)$ which can be computed as $P_i(t_1) = (1-k_{i,R})^{\delta_i(n_i)}\left(P_i^0-P_{i,\min}\right)+P_{i,\min}$, which is the initial value for the time interval $k\in[\alpha_i(n_i)+\delta_i(n_i), \alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)-1]$. With $u_i(k)=1$ for $k\in[\alpha_i(n_i)+\delta_i(n_i), \alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)-1]$, the dynamic of robot i's performance value is $P_i(k) = (1-k_{i,H})P_i(k-1)+k_{i,H}P_{i,\max}$. Then for any $k\in[\alpha_i(n_i)+\delta_i(n_i)+1, \alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)]$, we have $P_i(k) = (1-k_{i,H})^{k-t_1}P_i^1 + k_{i,H}P_{i,\max}\sum_{\kappa=t_1}^{k-1}(1-k_{i,H})^{k-1-\kappa} = (1-k_{i,H})^{k-t_1}\left(P_i^1-P_{i,\max}\right)+P_{i,\max}$. The costs among the time interval $[\alpha_i(n_i)+\delta_i(n_i)+1, \alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)]$ is $J^2_{i,n_i}(\delta_i(n_i)) = \sum_{k=\alpha_i(n_i)+\delta_i(n_i)+1}^{\alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)} P_i(k) = \dfrac{1-k_{i,H}}{k_{i,H}}\left[1-(1-k_{i,H})^{C_i(n_i)}\right]\left(P_i^1-P_{i,\max}\right)+C_i(n_i)P_{i,\max}$.

Let t_2 denote the time step $\alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)$ and P_i^2 denote $P_i(t_2) = (1-k_{i,H})^{C_i(n_i)}\left(P_i^1-P_{i,\max}\right)+P_{i,\max} = (1-k_{i,H})^{C_i(n_i)}(1-k_{i,R})^{\delta_i(n_i)}\left(P_i^0-P_{i,\min}\right)+(1-k_{i,H})^{C_i(n_i)}(P_{i,\min}-P_{i,\max})+P_{i,\max}$. For $k\in[\alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)+1, \alpha_i(n_i+1)-1]$, we have $u_i(k) = 0$, which leads to $P_i(k) = (1-k_{i,R})^{k-t_2}\left(P_i^2-P_{i,\min}\right)+P_{i,\min}$. The costs among the time interval $[\alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)+1, \alpha_i(n_i+1)-1]$ is $J^3_{i,n_i}(\delta_i(n_i)) = \sum_{k=\alpha_i(n_i)+\delta_i(n_i)+C_i(n_i)+1}^{\alpha_i(n_i+1)-1} P_i(k) = (t_3-\delta_i(n_i))P_{i,\min} + \dfrac{1-k_{i,R}}{k_{i,R}}\left[1-(1-k_{i,R})^{t_3-\delta_i(n_i)}\right]\left(P_i^2-P_{i,\min}\right)$. where $t_3 = T_i-1-C_i(n_i)$. If we denote $a_i = \frac{1-k_{i,R}}{k_{i,R}} > 0$, $b_i = \frac{1-k_{i,H}}{k_{i,H}} > 0$, $c_i = P_i^0-P_{i,\min}\ge 0$, $d_i = (1-k_{i,R})^{C_i(n_i)} > 0$, $e_i = (1-k_{i,H})^{C_i(n_i)} > 0$, $x_i = \ln(1-k_{i,R}) < 0$, $y_i = (1-k_{i,R})^{\delta_i(n_i)} > 0$ and then take the derivative of $J_{i,n_i}(\delta_i(n_i))$, we have $\frac{dJ_{i,n_i}(\delta_i(n_i))}{d\delta_i(n_i))} = (b_i-a_i)(1-e_i)c_ix_iy_i + a_i(e_i-1)x_i(1-k_{i,R})^{t_3-\delta_i(n_i)}(P_{i,\min}-P_{i,\max})$. Since $k_{i,H} < k_{i,R}$, we have $b_i-a_i > 0$. And it is trivial that $e_i-1 < 0$, therefore $(b_i-a_i)(1-e_i)c_ix_iy_i \le 0$ and $a_i(e_i-1)x_i(1-k_{i,R})^{t_3-\delta_i(n_i)}(P_{i,\min}-P_{i,\max})<0$, from which we can conclude that $\frac{dJ_{i,n_i}(\delta_i(n_i))}{d\delta_i(n_i))} < 0$. Therefore, when $\delta_i(n_i)=0$, $J_i(\delta_i(n_i))=J_i^{\max}$, which leads to (7). □

Based on Theorem 1, when there is no contention among robots, the optimal solution for $u_i(k)$ is

$$u_i(k)=\begin{cases}1, k\in[\alpha_i(n_i), \alpha_i(n_i)+C_i(n_i)-1],\\ 0, k\in[\alpha_i(n_i)+C_i(n_i), \alpha_i(n_i+1)-1],\end{cases} \quad (7)$$

for all n_i such that $k_0\le\alpha_i(n_i)$ and $\alpha_i(n_i+1)\le k_f$. And it is trivial that $\gamma_i(n_i)=\alpha_i(n_i)+C_i(n_i)$. However, if a contention occurs when robot i starts the request to collaborate with the human, then constraint (4) cannot be relaxed and equality $\gamma_i(n_i)=\alpha_i(n_i)+C_i(n_i)$ will not hold because it may be delayed by the collaboration between the human and other robots, i.e. $\delta_i(n_i)\ne 0$. And from the proof of Theorem 1, we know that the derivative $\frac{dJ_{i,n_i}(\delta_i(n_i))}{d\delta_i(n_i))}$ is strictly less than 0, which means the larger time delay $\delta_i(n_i)$ will further increase the cost in (5). Therefore, we need a timing model to compute the minimal value which the time delay variable $\delta_i(n_i)$ can take without violating the contention constraint. This CIA property also ensures that the human attention scheduling problem is analogous to the classic real-time scheduling problems.

III. Analytical Timing Model

Due to the constraint of the human attention, the collaboration request generation times $\alpha_i(n_i)$, the time instants when robots starts to collaborate with human $\alpha_i(n_i)+\delta_i(n_i)$, and the collaboration completion times $\gamma_i(n_i)$ are more significant than other moments. To obtain these significant moments, it is important to compute each $\delta_i(n_i)$, which is not easy to compute. We leverage scheduling theory and model the human attention scheduling as one classical scheduling discipline, the non-preemptive scheduling discipline. In scheduling theory [15], if an on-going task (with process time $C_i(n_i)$ can be interrupted by the generation of other tasks, the scheduling type is called *preemptive*. In work [9], the collaboration tasks are modeled as preemptive tasks on a single processor, which can results in a schedule where the human operator needs to constantly switch among different robots and can increase difficulty for the human to work with robots. Therefore, in this paper, we add in a constraint that the collaboration between a robot and a human is *non-preemptive*, where a non-preemptive task means that an on-going task cannot be interrupted by the generation of other tasks, to reduce the workload of the human operator.

A. Timing States

In [9], [16], we developed a Significant Moment Analysis (SMA) method and timing models that can compute the time delays in the preemptive model. In [17], a continuous-time timing model was developed for non-preemptive model. Here we derive a discrete-time analytical timing model.

At each time k on time horizon $[k_0, k_f]$ of the optimization problem, we define the timing state variable $Z(k) = (D(k), R(k), O(k), \mathrm{ID}(k))$ as follows.

Definition 1: The deadline variable is $D(k) = (d_1(k), ..., d_i(k), ..., d_N(k))$, where $d_i(k)$ is defined to be how long after time k the next generation time $\alpha_i(n_i)$ of task τ_{i,n_i} will be generated. The remaining time variable is $R(k) = (r_1(k), ..., r_i(k), ..., r_N(k))$, where $r_i(k)$ denotes the remaining collaboration time after time k that is needed to complete the collaboration of the most recently generated task τ_{i,n_i}. The delay variable is $O(k) = (o_1(k), ..., o_i(k), ..., o_N(k))$, where $o_i(k)$ is how long the collaboration completion of task τ_{i,n_i} has been delayed from its most recent request time $\alpha_i(n_i)$ to time k. The index variable $\mathrm{ID}(k)$ is index of the robot that is collaborating with the human operator at time k, where $\mathrm{ID}(k)\neq 0$ implies that the human attention is occupied by one robot and $\mathrm{ID}(k)=0$ implies that no robot is occupying the human attention at time k.

To support the dynamic timing model, we redefine the collaboration time of a task as follows:

Definition 2: For all i, $n_i\geq 0$, and $k\in[\alpha_i(n_i), \alpha_i(n_i+1)]$, we set $C_i(k)=C_i(n_i)$ for $k\in[\alpha_i(n_i), \alpha_i(n_i+1))$.

The evolution rules for $Z(k)$ can be expressed by an analytical model that is efficient to compute which supports the implementation of contention-resolving MPC.

B. Timing Model for Human Attention Scheduling

We divide the scheduling time horizon $[k_0, k_f]$ into sub-intervals $[k_w, k_{w+1}]$ such that $k_{w+1} - k_w = \mathrm{sgn}(\mathrm{ID}(k_w)) \min\{r_{\mathrm{ID}}(k_w), d_1(k_w), ..., d_N(k_w), k_f - k_w\} + (1-\mathrm{sgn}(\mathrm{ID}(k_w))) \min\{d_1(k_w), ..., d_N(k_w), k_f - k_w\}$ for all w, where $r_{\mathrm{ID}}(k)$ is a simplified notation for the remaining time $r_{\mathrm{ID}(k)}(k)$ of robot $\mathrm{ID}(k)$ at time k.

At time k_w, if $r_{\mathrm{ID}}(k_w-1) > 1$, which means the robot $\mathrm{ID}(k_w-1)$ that was occupying the human attention has not completed the collaboration at time k_w, then $\mathrm{ID}(k_w)$ is the same as $\mathrm{ID}(k_w-1)$ because the collaboration is non-preemptive. If $r_{\mathrm{ID}}(k_w-1) = 1$, which means the robot $\mathrm{ID}(k_w-1)$ completed the collaboration at time k_w, then $\mathrm{ID}(k_w)$ needs to switch to the robot which is scheduled to collaborate with the human operator, i.e. the robot i with $u_i(k_w)=1$. Combining these two cases, the evolution rule for the timing state ID can be expressed as $\mathrm{ID}(k_w) = \mathrm{ID}(k_w-1)\,\mathrm{sgn}(r_{\mathrm{ID}}(k_w-1)-1) + \mathrm{argmax}_i\,\{u_i(k_w)\}\big[1-\mathrm{sgn}(r_{\mathrm{ID}}(k_w-1)-1)\big]$ where $\mathrm{sgn}(q) = 1$ if $q > 0$ and $\mathrm{sgn}(q)=0$ if $q=0$. If $\sum_{i=1}^{N} u_i(k_w)=0$, then $\mathrm{ID}(k_w)=0$.

At time k_w, the values of the variables d_i, r_i and o_i have jumps for some i. If the deadline variable of robot i satisfies $d_i(k_w-1) = 1$, then $d_i(k_w) = T_i, r_i(k_w) = C_i(k_w)$ and $o_i(k_w) = 0$. If $d_i(k_w-1) > 1$, then there are no jumps for the timing states for robot i and we have $d_i(k_w)=d_i(k_w-1)-1$, $r_i(k_w)=r_i(k_w-1)-\mathbb{1}(\mathrm{ID}(k_w-1)=i)$ and $o_i(k_w)=o_i(k_w-1)+\mathrm{sgn}(r_i(k_w-1))$ where $\mathbb{1}(\cdot)$ is an indicator function which is defined to be 1 if the condition $\mathrm{ID}(k_w-1)=i$ holds and 0 otherwise. Combining the two cases depending on the different values of $d_i(k_w-1)$, the evolution rules of the timing state variables d_i, r_i and o_i at the times k_w can be summarized as $d_i(k_w)=d_i(k_w-1)-1+\big[1-\mathrm{sgn}(d_i(k_w-1)-1)\big]T_i$, $r_i(k_w)=\mathrm{sgn}(d_i(k_w-1)-1)\big[r_i(k_w-1)-\mathbb{1}(\mathrm{ID}(k_w-1)=i)\big]+\big[1-\mathrm{sgn}(d_i(k_w-1)-1)\big]C_i(k_w)$ and $o_i(k_w)=\big[o_i(k_w-1)+\mathrm{sgn}(r_i(k_w-1))\big]\,\mathrm{sgn}(d_i(k_w-1)-1)$.

During any time $k_w+\Delta k\in[k_w+1, k_{w+1}-1]$, the state $\mathrm{ID}(k_w+\Delta k)$ remains unchanged because $k_{w+1}-k_w \leq r_{\mathrm{ID}}(k_w)$. If $ID(k_w)\neq 0$, the evolution rules for robot $ID(k_w)$ are $d_{\mathrm{ID}}(k_w+\Delta k)=d_{\mathrm{ID}}(k_w)-\Delta k$, $r_{\mathrm{ID}}(k_w+\Delta k)=r_{\mathrm{ID}}(k_w)-\Delta k$ and $o_{\mathrm{ID}}(t_w+\Delta t) = o_{\mathrm{ID}}(t_w)+\Delta t$ where $d_{\mathrm{ID}}(k)$ and $o_{\mathrm{ID}}(k)$ are defined analogously to $r_{\mathrm{ID}}(k)$. And during this time window, we have $u_{\mathrm{ID}}(k_w+\Delta k)=1$. For robot $i\neq ID(k_w)$, the evolution rules are $d_i(k_w+\Delta k) = d_i(k_w)-\Delta k$, $r_i(k_w+\Delta k) = r_i(k_w)$ and $o_i(k_w+\Delta k) = o_i(k_w)+\mathrm{sgn}(r_i(k_w))\Delta k$. During this time window, for robot $i\neq ID(k_w)$, we have $u_i(k_w+\Delta k)=0$.

Combining all of the evolution rules above leads to the timing model of non-preemptive human attention scheduling, which provides the value of $Z(k)$ at each time k, given the initial state variable $Z(k_0)$, the vehicle timing parameters $C_i(n_i)$ and T_i for all i and n_i, and the value of $\mathbf{u}(k_0\sim k)$, where $\mathbf{u}(k_0\sim k)$ is a simplified notation for the decision variable $\mathbf{u}$ at all time steps in the interval $[k_0, k]$

$$Z(k)=\mathbb{H}(k; Z(k_0), (C_i(n_i), T_i)_{i=1,...,N}, \mathbf{u}(k_0\sim k)). \quad (8)$$

Then the definition of the variable $O(k)$ gives $\delta_i(n_i) = o_i(\alpha_i(n_i+1)-1)+\mathrm{sgn}(r_i(\alpha_i(n_i+1-1))-C_i(n_i)$.

Remark 2: Notice that the only times when the values of timing state depend on the decision variable u_i are at the significant moments k_w. Therefore, we only need to determine the value of u_i at those times.

IV. Contention-Resolving MPC Algorithm

In this section, we consider the original problem formulation where constraints (3) and (4) are not relaxed. We convert the problem formulated by (5) into a path planning problem that can be solved iteratively. The conversion is based on the insight that value of the decision variable $\mathbf{u}$ only needs to be decided at the significant moments when contention occurs.

A. Construction of Decision Tree

We use the timing model to determine when contentions occur by checking the following condition:

Proposition 1: A contention starts at time k if and only if the following three conditions hold 1) $\sum_{i=1}^{N}\left[1-\operatorname{sgn}(C_i(k)-r_i(k))\right]\geq 2$, 2) $r_{\mathrm{ID}}(k-1)\leq 1$ and 3) $k=k_w$ for some i and some w where k_w is a significant moment computed by (8).

Proof. A collaboration request from robot i is waiting at a time k or is generated at time k if and only if $r_i(k) = C_i(k)$, i.e., $1-\operatorname{sgn}(C_i(k)-r_i(k)) = 1$. Therefore, $\sum_{i=1}^{N}\left[1-\operatorname{sgn}(C_i(k)-r_i(k))\right]\geq 2$ if and only if two or more robots are waiting to collaborate with the human operator at time k or generating requests at time k. Therefore, if a contention starts at time k, then k is one significant moment k_w for some i and w. And the robot that was collaborating with the human one time step before k will either finish the collaboration at time k, i.e., $r_{\mathrm{ID}}(k-1)=1$ or the human was not collaborating with any robot one time step before k, i.e., $r_{\mathrm{ID}}(k-1)=0$, so $r_{\mathrm{ID}}(k-1)\leq 1$ holds. Conversely, if the three conditions are satisfied, then at time k, multiple robots are in contention to collaborate with the human which is a necessary condition, so a contention starts at time k. □

Based on the contention times, we can construct a decision tree. Figure 1 shows an example of decision tree. In the decision tree, each leaf represents a contention time satisfying Proposition 1. We denote the contention times by k_l^c where l is the index of its corresponding leaf. The construction of the entire decision tree is not necessary for the contention resolving MPC algorithm. However, for the purposes of clearly presenting the sampling based optimization method, we now briefly describe how the tree can be fully constructed.

The decision tree construction starts from the root v_0 associated with the MPC starting time k_0. The construction is performed iteratively. During the construction, if a leaf has no branches pointing out from it, then it is called *unexpanded*. At each iteration, new branches are generated from unexpanded leaves and new leaves are generated at the end of each branch. For an unexpanded leaf l, let $\Lambda\left(k_l^c\right)$ denote the set of indices of robots having contentions at a contention time k_l^c in an increasing order. We also define M to be the number of elements in $\Lambda\left(k_l^c\right)$. For leaf l, we generate M branches from it. Each branch corresponds to a unique vector $\mathbf{u}_m\left(k_l^c\right)=\left(u_{m,1}\left(k_l^c\right),u_{m,2}\left(k_l^c\right),\dots,u_{m,N}\left(k_l^c\right)\right)$. Let i be the

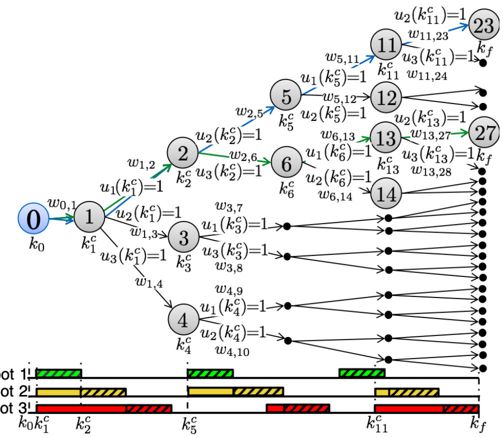

Fig. 1. Decision tree to solve the integer optimization problem for a finite time window. The blue circle represents the root v_0, and grey circles and dots represent leaves. The decision tree is expanded in the direction of the arrows, which represent the branches. The colored rectangles in the lower part of the figure represent the time delay δ_i. The starting time of the colored rectangles is the request generation time α. The shaded colored rectangles represent the human attention occupation time of each robot.

mth element in set $\Lambda\left(k_l^c\right)$. Then in vector $\mathbf{u}_m\left(k_l^c\right)$, we define $u_{m,i}\left(k_l^c\right)=1$ and $u_{m,q}\left(k_l^c\right)=0$ for all $q=1,\dots,N$ and $q\neq i$. The mth branch expands from v_l and connects to a new leaf v_{j+m} based on $\mathbf{u}_m$, where j is the number of existing leaves in the tree before we generate new branches from leaf v_l. The contention time represented by leaf v_{j+m} is the next contention time that occurs after k_l^c as scheduled by $\mathbf{u}_m$. The iterative construction terminates when the contention times of all unexpanded leaves are greater or equal to k_f. Then we call these unexpanded leaves *terminal leaves*.

B. Branch Cost

After constructing the decision tree, we define a cost for each branch. Along one branch (l,j) connecting leaves v_l and v_j, since the decision variables $\mathbf{u}(k)$ are determined for all i and $k\in[k_l^c,k_j^c]$, we can calculate the significant moments $\gamma_i(n_i)$ for all i and n_i such that $k_l^c\leq\gamma_i(n_i)\leq k_j^c$ as $Z(k)=\mathbb{H}\Big(k;Z(k_l^c),(C_i(n_i),T_i)_{i=1,\dots,N},\mathbf{u}_m\Big)$ and $\gamma_i(n_i)=\alpha_i(n_i)+o_i(\alpha_i(n_i+1)-1)+\operatorname{sgn}(r_i(\alpha_i(n_i+1-1))$ where $r_i(\alpha_i(n_i+1)-1)$ and $o_i(\alpha_i(n_i+1)-1)$ for each n_i are generated by the timing model except with a known $\mathbf{u}_m$. Then the branch cost $w_{l,j}$ is defined as $w_{l,j}=\sum_{i=1}^{N}w_{l,j}^i$ where $w_{l,j}^i$ is the cost of robot i. For each i such that there is a completion time $\gamma_i(n_i{+}1)\in(t_l^c,t_j^c]$, let $\underline{n}_i$ be the smallest index n_i satisfying $\gamma_i(n_i{+}1)>k_l^c$ and $\overline{n}_i$ be the largest index n_i satisfying $\gamma_i(n_i{+}1)\leq k_j^c$. Then we set

$$w_{l,j}^i=\sum_{k=\gamma_i(\underline{n}_i)}^{\gamma_i(\overline{n}_i)-1}\left[P_{i,\max}-P_i(k)\right],\ k\in[\gamma_i(\underline{n}_i),\gamma_i(\overline{n}_i)-1]. \quad (9)$$

If no collaboration of robot i is completed within $[t_l^c,t_j^c]$, i.e. $\underline{n}_i>\overline{n}_i$, then we define $w_{l,j}=0$. And if for any time $k\in[t_l^c,t_j^c]$, we have $\mathcal{T}_i(k)<\mathcal{T}_{i,\min}$ or $\mathcal{T}_i(k)>\mathcal{T}_{i,\max}$, then we define $w_{l,j}=+\infty$. The meaning of (9) is as follows. If $\gamma_i(n_i)\in(k_l^c,k_j^c]$, i.e., the n_ith collaboration of robot i is completed between the contention times k_l^c and k_j^c, then the cost of the n_ith task is included in the branch cost $w_{l,j}$. This branch cost formulation ensures that all costs included in one branch are determined and will not be changed by the decision variable $\mathbf{u}$ at or after time k_j^c. The cost of an uncompleted $(\overline{n}_i+1)$st collaboration will be included by the branches following the branch (l, j). Based on the decision tree, the integer optimization problem in (5) can now be converted to the problem of finding a path from k_0 to k_f such that the whole cost along the path is lowest. In our previous work [12], we presented a contention-resolving MPC algorithm that leverages the A-star algorithm to search for an optimal path in the decision tree. We define the same stage cost $g(v_l)$ to be the same as [12]. And the heuristic future cost $\hat{h}(v_l)$ to be $\hat{h}(v_l) = \sum_{i=1}^{N} \sum_{k=\gamma_i(\overline{n}_i)}^{k_f} [P_{i,\max} - P_i(k)]$ with $u_i(k)$ satisfying (7) for all n_i such that $k_l^c \leq \alpha_i(n_i)$ and $\alpha_i(n_i+1) \leq k_f$. This cost considers cases without contention constraints and is less than or equal to the true future with contention constraints, as we explained in Theorem 1. We have showed in [12] that the minimal cost path is guaranteed to be found with these defined costs. The search algorithm only efficiently generates a subtree without losing optimality.

V. Simulation Results

We simulate three robots collaborating with one human operator. The starting and ending time instants are $k_0 = 0$ and $k_f = 120$ respectively. The initial values of trust level are $[\mathcal{T}_1(0), \mathcal{T}_2(0), \mathcal{T}_3(0)] = [1.93, 1.9, 1.98]$. The lower and upper bounds for trust level are $[\mathcal{T}_{1,\min}, \mathcal{T}_{2,\min}, \mathcal{T}_{3,\min}] = [1.55, 1.65, 1.7]$ and $[\mathcal{T}_{1,\max}, \mathcal{T}_{2,\max}, \mathcal{T}_{3,\max}] = [2.15, 2.35, 2.1]$. The parameters for trust model are $A_i = 1$, $B_i = 0.605$ and $C_i = 0.6$ for all i. The initial values of robot performance are $[P_1^0(n_i), P_2^0(n_i), P_3^0(n_i)] = [0.7, 0.7, 0.7]$ for all n_i. The parameters for the robot performance model are $[k_{1,R}, k_{2,R}, k_{3,R}] = [0.25, 0.25, 0.25]$ and $[k_{1,H}, k_{2,H}, k_{3,H}] = [0.1, 0.13, 0.15]$. The lower and upper bounds are $[P_{1,\min}, P_{2,\min}, P_{3,\min}] = [0.6, 0.65, 0.65]$ and $[P_{1,\max}, P_{2,\max}, P_{3,\max}] = [0.75, 0.75, 0.75]$. The timing parameters are $[C_1(n_i), C_2(n_i), C_3(n_i)] = [6, 6, 6]$ for all n_i and $[T_1, T_2, T_3] = [20, 30, 40]$. The human attention occupation result is shown in Figure 2. The robot performance is shown in Figure 3. Five contentions occur in the time interval $[0, 120]$. The cost under optimal schedule is 31.4262, which is 11.99% less than the HTS scheduling. The simulation results show that our method performs better than HTS.

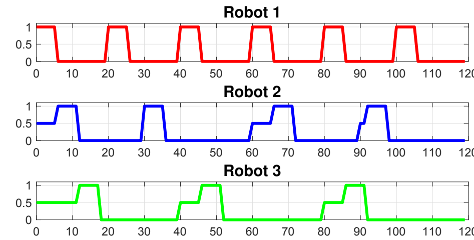

Fig. 2. Human attention occupation for collaborating with three robots. The y axis value 1 means that the robot is collaborating witht he human, 0 means that the robot is not requesting the collaboration, and 0.5 means that the robot's collaboration request is delayed by a contention.

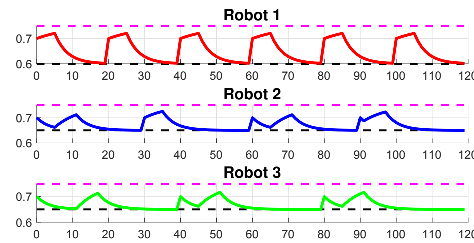

Fig. 3. Performance of three robots under the optimal schedule. The magenta dashed line represents $P_{i,\max}$ and the black line represents $P_{i,\min}$.

Acknowledgement

The authors thank Yuki Gao for her help to this work.

References

[1] M. L. Cummings, J. P. How, A. Whitten, and O. Toupet, "The impact of human–automation collaboration in decentralized multiple unmanned vehicle control," *Proceedings of the IEEE*, vol. 100, no. 3, pp. 660–671, 2011.

[2] A. Kolling, P. Walker, N. Chakraborty, K. Sycara, and M. Lewis, "Human interaction with robot swarms: A survey," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 1, pp. 9–26, 2015.

[3] G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information." *Psychological review*, vol. 63, no. 2, p. 81, 1956.

[4] N. Cowan, "The magical mystery four: How is working memory capacity limited, and why?" *Current directions in psychological science*, vol. 19, no. 1, pp. 51–57, 2010.

[5] N. Carr, *The shallows: What the Internet is doing to our brains*. WW Norton & Company, 2011.

[6] C. L. Liu and J. W. Layland, "Scheduling algorithms for multiprogramming in a hard-real-time environment," *Journal of the ACM*, vol. 20, no. 1, pp. 46–61, 1973.

[7] J. W. Crandall, M. L. Cummings, M. Della Penna, and P. M. De Jong, "Computing the effects of operator attention allocation in human control of multiple robots," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 41, no. 3, pp. 385–397, 2010.

[8] S.-Y. Chien, Y.-L. Lin, P.-J. Lee, S. Han, M. Lewis, and K. Sycara, "Attention allocation for human multi-robot control: Cognitive analysis based on behavior data and hidden states," *International Journal of Human-Computer Studies*, vol. 117, pp. 30–44, 2018.

[9] X. Wang, Z. Shi, F. Zhang, and Y. Wang, "Dynamic real-time scheduling for human-agent collaboration systems based on mutual trust," *Cyber-Physical Systems*, vol. 1, no. 2-4, pp. 76–90, 2015.

[10] J. W. Crandall, M. A. Goodrich, C. W. Nielsen, and D. R. Olsen Jr, "Validating human–robot interaction schemes in multitasking environments," 2005.

[11] C. C. Murray and W. Park, "Incorporating human factor considerations in unmanned aerial vehicle routing," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 4, pp. 860–874, 2012.

[12] N. Yao, M. Malisoff, and F. Zhang, "Contention resolving optimal priority assignment for event-triggered model predictive controllers," in *American Control Conference (ACC), 2017*. IEEE, 2017, pp. 2357–2362.

[13] N. Yao and F. Zhang, "Resolving contentions for intelligent traffic intersections using optimal priority assignment and model predictive control," in *2018 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2018, pp. 632–637.

[14] S.-Y. Chien, M. Lewis, S. Mehrotra, N. Brooks, and K. Sycara, "Scheduling operator attention for multi-robot control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 473–479.

[15] R. W. Conway, W. L. Maxwell, and L. W. Miller, *Theory of Scheduling*. Courier Corporation, 2003.

[16] Z. Shi and F. Zhang, "Model predictive control under timing constraints induced by controller area networks," *Real-Time Systems*, pp. 1–32, 2015.

[17] ——, "Model predictive control under timing constraints induced by controller area networks," *Real-Time Systems*, vol. 53, no. 2, pp. 196–227, 2017.