

User-interface design for MIMO LTI human-automation systems through sensor placement

Abraham P. Vinod, Tyler H. Summers, and Meeko M.K. Oishi

Abstract—We propose a method for user-interface design for a MIMO LTI human-automation system by solving a corresponding sensor placement problem. We derive suitable metrics for optimization by transforming human factors guidelines for human-automation interaction into objective functions and constraints. We consider operation under 1) both nominal and off-nominal conditions, which have vastly different user requirements, and 2) noisy and noise-free sensors. The resulting optimization problems are combinatorial. We analyze optimal solutions, and provide algorithms to construct sub-optimal solutions when optimal solutions are not readily available. We apply this method to the problem of designing a user-interface for the remote operation of a fleet of UAVs.

Index Terms—User-interface design, human-automation interaction, observability, sensor placement, output synthesis

I. INTRODUCTION

Problems in human-automation interaction have contributed to major failures in expensive, high-risk, and safety-critical systems (including aircraft and other transportation incidents and accidents, overdosage via biomedical devices, power grid distribution systems, nuclear power generation). A key element of any human-automation system is the user-interface, which provides information about the system to the user. Large human-automation systems are prohibitively complex for intuitive design of user-interfaces. While certain aspects of user-interface design are rightly in the domain of human factors, engineering psychology, and human-computer interaction, the information content of the user-interface should be dynamics driven. We consider the user-interface to be an output map of a dynamical system, and pose the question of user-interface design as one of *sensor placement*: we aim to identify a reduced representation of the system which is optimal with respect to well established human factors design constraints for “good” human-automation interaction. These design constraints could include operational conditions (e.g., well known, nominal scenarios or rarely encountered, off-nominal scenarios), the user’s situational awareness and trust of the automation, and human-computer interaction guidelines (e.g., not providing

too little or too much information in the user-interface), for example.

User-interface design has historically been approached via human-computer interaction heuristics [1], however the need for formal methods is well established in aerospace [2] and other application domains [3]. For discrete event systems (or discrete abstractions of hybrid systems), model-checking can provide guidance for redesign of faulty interfaces for discrete event systems [4], [5], [6], and finite-state machine reduction [7], [8] has been used to synthesize user-interfaces of minimal cardinality. Our prior work has addressed analysis of an existing interface for a (hybrid) LTI system under shared control [9], [10], [11]. However, no methodological approaches exist for *design* of user-interfaces for continuous or hybrid dynamical systems.

User-interface design can be interpreted as a problem of sensor selection. Sensor selection problems arise in many applications, including robotics [12], [13], wireless surveillance networks [14], and power systems [15], [16], and can be posed as combinatorial optimization problems. Research on sensor selection often focuses on identifying appropriate metrics for distinguishing a given selection [17], [18]. Finding an optimal selection for a specific metric is challenging and in general requires exhaustive search, which quickly becomes intractable even for moderate problem sizes. The main heuristics are convex relaxation [19], [20] and purely combinatorial algorithms that avoid a full exhaustive search [21], [22], [23]. Convex relaxation approaches utilize sparsity inducing regularizers (e.g., ℓ_1 norms) to obtain sparse designs from continuous variables that have relaxed binary choices. Purely combinatorial approaches utilize simple greedy algorithms and other graph theoretic approaches [22], [23], [24]. For some problem classes (e.g., cardinality-constrained submodular set function maximization [21], [25]), these algorithms yield provably optimal or near-optimal sensor selections. However, none of the methodological approaches for sensor selection consider criteria and constraints for human-automation interaction.

The main contribution of this paper is user-interface *synthesis* for human-automation systems under a “manual” control configuration in which the user provides the high-level input commands, and the automation synthesizes the corresponding low-level actuator commands [10]. We focus on MIMO LTI dynamical systems with and without measurement noise and pose the problem of user-interface design as one of the selection of an output map that is optimal with respect to certain design constraints that represent “good” human-automation interaction. The solution is based

This material is based upon work supported by the National Science Foundation. Oishi and Vinod are supported under Grant Number CMMI-1254990 (CAREER, Oishi), CNS-1329878, and CMMI-1335038. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Abraham P. Vinod and Meeko Oishi (corresponding author) are with Electrical and Computer Engineering, University of New Mexico, Albuquerque, NM 87131 USA; e-mail: aby.vinod@gmail.com, oishi@unm.edu

Tyler H. Summers is with Mechanical Engineering, University of Texas at Dallas, Richardson, TX 75080; e-mail: tyler.summers@utdallas.edu

on methods for sensor placement, but since the problems formulated in this paper do not map easily to standard algorithms, we characterize the sets in which the optimal solution is guaranteed to lie.

The paper is organized as follows: We define the problem and the design constraints for user-interface design in Section II. Section III formulates the four user-interface design problems (nominal/off-nominal and with/without measurement noise) as combinatorial optimization problems. We apply our methods to an example of remote control of a fleet of UAVs in Section IV, and provide conclusions and directions for future work in Section V.

II. PROBLEM FORMULATION

Consider the dynamical system,

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= C_S x(t) + v(t) \end{aligned} \quad (1)$$

with state $x(t) \in \mathbb{R}^n$, input $u(t) \in \mathbb{R}^m$, output $y(t) \in \mathbb{R}^p$, measurement noise $v(t) \in L_2^p([0, \infty))$ with finite energy $\|v(t)\|_2 \leq V^2$, and matrices $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times m}$.

We consider a finite set of N sensors $\mathcal{S} = \{s_1, \dots, s_N\}$, with the i^{th} sensor denoted as $s_i \in \mathbb{R}^n$. That is, each sensor is represented as a measurement which is a linear combination of states. Then the set of possible sensor combinations is the power set $2^{\mathcal{S}}$. The output matrix $C_S \in \mathbb{R}^{p \times n}$ is a matrix whose rows consist of the $p = |\mathcal{S}|$ elements of the sensor combination $S \in 2^{\mathcal{S}}$.

Without loss of generality, B is assumed to have full column rank and $m, p \leq n$. For the case in which there is no measurement noise $v(t)$, system (1) reduces to

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= C_S x(t) \end{aligned} \quad (2)$$

As in [9], [10], we make the following assumptions:

Assumption 1. *The user understands the outputs of the system and can comprehend its higher derivatives.*

Assumption 2. *The automation is designed correctly and will track a given reference trajectory.*

Assumption 1 is an interpretation of situational awareness, a key element of effective human-automation interaction [26], [27]. This assumption suggests that an input-output model of (2) can capture the user's interaction with the system, as both an observer and controller. This assumption also guides the development of optimization metrics presented in Section III. For systems with relative degree smaller than the order of the system, the transformation to an input-output model will generate zero dynamics. Assumption 2 helps in establishing the stability of the zero dynamics, since the stability of a system is invariant under similarity transformation.

Let $\rho(M)$ represent the rank of a matrix M , $\mathcal{N}(M)$ be the null space of M , $\mathcal{R}^\perp(M)$ be the row space of M , and $\text{basis}(\mathcal{R}^\perp(M))$ be a matrix-valued function which removes the linearly dependent rows of M . The result of this operation is not unique.

A. Transformation to normal form

For an output matrix C_S defined using $S = \{s_1, s_2, \dots, s_{|S|}\} \in 2^{\mathcal{S}}$ where $|S|$ is the cardinality of S , we transform the system (2) to normal form

$$\begin{bmatrix} \xi(t) \\ \eta(t) \end{bmatrix} = \begin{bmatrix} T(S) \\ T^\perp(S) \end{bmatrix} x(t), \quad T(S) \in \mathbb{R}^{\gamma(S) \times n} \quad (3)$$

where by definition, $[T(S)^\top (T^\perp(S))^\top] \in \mathbb{R}^{n \times n}$ is non-singular [28], [29]. The relative degree of the system $\gamma(\cdot) : 2^{\mathcal{S}} \rightarrow \{1, 2, \dots, n\}$ is a set function,

$$\gamma(S) = \rho(T(S)). \quad (4)$$

For the system (2), the transformation given in (3) gives

$$\begin{aligned} \dot{\xi}(t) &= A_1 \xi(t) + A_2 \eta(t) + B_m u(t) \\ \dot{\eta}(t) &= A_\xi \xi(t) + A_\eta \eta(t) \\ y(t) &= I_{p \times \gamma(S)} \xi(t) \end{aligned} \quad (5)$$

with matrices $A_1 \in \mathbb{R}^{\gamma(S) \times \gamma(S)}$, $A_2 \in \mathbb{R}^{\gamma(S) \times (n - \gamma(S))}$, $B_m = T(S)B \in \mathbb{R}^{\gamma(S) \times m}$, $A_\xi \in \mathbb{R}^{(n - \gamma(S)) \times \gamma(S)}$ and $A_\eta \in \mathbb{R}^{(n - \gamma(S)) \times (n - \gamma(S))}$. Here, the linear transformation $T(S)$ is defined such that $T(S) = \text{basis}(\mathcal{R}^\perp([T(s_1)^\top T(s_2)^\top \dots T(s_{|S|})^\top]^\top))$ and $T(s_i) = \left[s_i (s_i^\top A)^\top (s_i^\top A^2)^\top \dots (s_i^\top A^{\gamma(s_i) - 1})^\top \right]^\top$ for every $s_i \in S$. Also, $\gamma(s_i)$ is the relative degree associated with an output channel $y_i(t) = s_i^\top x(t)$, and $\gamma(s)$ for any sensor $s \in S$ can be computed using Markov parameters. The linear transformation $T^\perp(S)$ is defined such that $T^\perp(S)B = 0$ [29].

By definition, $T(S)$ is the basis of concatenated similarity transforms corresponding to each $s_i \in S$. If C_S has full row rank, we can enforce, without loss of generality, the constraint that the first $|S|$ rows of $T(S)$ are C_S since $\text{basis}(\cdot)$ is not unique. This allows for the definition of $y(t)$ in (5). Note that the transformation of (1) results in similar dynamics, with extra terms due to noise $v(t)$.

For a multi-output system, let $\gamma^t(S) = \sum_{s_i \in S} \gamma(s_i)$, and by definition, $\gamma^t(S) \geq \gamma(S)$. We presume a differentiable reference trajectory $y_R(t) = [\tilde{y}_1(t) \tilde{y}_2(t) \dots \tilde{y}_{|S|}(t)]$ with $\tilde{y}_i(t)$ the reference trajectory for the i^{th} output $y_i(t) = s_i^\top x(t)$. We define $d\tilde{y}_i(t) = [\tilde{y}_i^{(1)}(t) \tilde{y}_i^{(2)}(t) \dots \tilde{y}_i^{(\gamma(s_i) - 1)}(t)]$ and $\xi_{\text{ref}}(t) = [y_R(t) d\tilde{y}_1(t) d\tilde{y}_2(t) \dots d\tilde{y}_{|S|}(t)]^\top$, with dynamics

$$\begin{aligned} \dot{\xi}_{\text{ref}}(t) &= A_{\text{ref}} \xi_{\text{ref}}(t) + B_{\text{ref}} \bar{y}_R(t) \\ \xi_R(t) &= C_{\text{ref}}(S) \xi_{\text{ref}}(t), \quad C_{\text{ref}}(S) \in \mathbb{R}^{\gamma(S) \times \gamma^t(S)} \end{aligned} \quad (6)$$

The matrix $C_{\text{ref}}(S)$ is defined such that $T(S) = C_{\text{ref}}(S)[T(s_1)^\top T(s_2)^\top \dots T(s_{|S|})^\top]$ and $\bar{y}_R(t)$ is the vector when the i^{th} term in $y_R(t)$ differentiated $\gamma(s_i)$ times. The matrices $A_{\text{ref}} \in \mathbb{R}^{\gamma^t(S) \times \gamma^t(S)}$ and $B_{\text{ref}} \in \mathbb{R}^{\gamma^t(S) \times p}$ are defined to have appropriate structure [10]. The output $\xi_R(t)$ serves as the reference trajectory for $\xi(t)$ when the human input is $y_R(t)$.

The external dynamics in (5) can be made to track $\xi_R(t)$ by solving a LQR tracking problem with $Q = \begin{bmatrix} I_{\gamma(S) \times \gamma(S)} & 0 \\ 0 & 0 \end{bmatrix}$ and $R = I_{m \times m}$ [30]. Hence, the low-level automation input is

$$u(t) = -K_\xi(t)(\xi(t) - \xi_R(t)) - K_\eta(t)\eta(t) + u_{fw}(t) \quad (7)$$

where $\xi_R(t)$ is determined from the high-level human input $y_R(t)$ using (6). Using (7) and defining $\xi_c(t) \triangleq B_m K_\xi(t) \xi_R(t) + B_m u_{fw}(t)$, the system (5) becomes

$$\begin{aligned} \dot{\xi}(t) &= (A_1 - B_m K_\xi(t)) \xi(t) + (A_2 - B_m K_\eta(t)) \eta(t) \\ &\quad + \xi_c(t) \\ \dot{\eta}(t) &= A_\xi \xi(t) + A_\eta \eta(t) \\ y(t) &= I_{p \times \gamma(S)} \xi(t) \end{aligned} \quad (8)$$

where $[K_\xi(t) \ K_\eta(t)] = B_M^\top P(t)$ and $P(t)$ is the time-varying solution to the corresponding Riccati equation. The feedforward signal $u_{fw}(t)$ is defined as $u_{fw}(t) = -B_M^\top b(t)$ for $b(t) = [\int_t^\infty \phi_b(t, \tau) P(\tau) C_{\text{ref}}(S) e^{A_{\text{ref}}(\tau-t)} d\tau] \xi_{\text{ref}}(t)$ with $\phi_b(t, \tau)$ defined as the state transition matrix for the time-varying system $\dot{b}(t) = -[A_M - B_M B_M^\top P(t)]^\top b(t)$ with matrices $A_M = \begin{bmatrix} A_1 & A_2 \\ A_\xi & A_\eta \end{bmatrix} \in \mathbb{R}^{n \times n}$ and $B_M = \begin{bmatrix} B_m \\ 0 \end{bmatrix} \in \mathbb{R}^{n \times m}$. By construction, $u_{fw}(t)$ and $\xi_c(t)$ can be computed causally [30].

The definition of $u(t)$ restricts the observable states of (2) to $\xi(t)$, and hence, the dimension of observable space is $\gamma(S)$ [28]. From Assumption 2 and the equivalence of the systems (2) and (5), it is evident that the zero dynamics $\eta(t)$ are stable as $\xi(t) \rightarrow \xi_R(t)$.

B. Task space and availability of a user interface

Tasks can be defined as a safety or liveness specifications [10]. We formulate the set of states for which the specified task is feasible as

$$\mathcal{F}(t) = \{x(t) : l(C_{\text{task}} x(t)) \geq 0\}, \quad C_{\text{task}} \in \mathbb{R}^{w \times n}$$

with the (possibly nonlinear) function $l : \mathbb{R}^w \rightarrow \mathbb{R}^\nu$ with ν subtasks, and the task matrix C_{task} is defined using a possible combination of sensor placements $S_{\text{task}} \in 2^S$. For tasks involving safety, the state $x(t)$ must lie in $\mathcal{F}(t)$ for all time in order for the task to be successfully completed. For liveness tasks, the task is successfully completed when $x(t)$ enters the set $\mathcal{F}(t)$ at some time. We denote the task space \mathbb{T} by the row space of C_{task} , i.e., $\mathbb{T} = \mathcal{R}^\perp(C_{\text{task}})$. Clearly, \mathbb{T} is the subset of the domain of l that the user needs to be aware of. Hence, the information required to complete the specified task (l, C_{task}) is completely characterized by S_{task} .

The user-interface to a system (2) under a ‘‘manual’’ control configuration is said to be *available* for a given task specified by S_{task} if the corresponding task space $\mathbb{T} \subseteq \mathcal{X}_O \cap \mathcal{X}_P$ where \mathcal{X}_O is the user-observable space and \mathcal{X}_P is the user-predictable space of (2), e.g., the states that the user can reconstruct and the state derivatives that the user can reconstruct, respectively, with information available to the

user [9]. This design constraint ensures that the user-interface provides the user with adequate information to successfully accomplish the task [10]. In short, these conditions assure that the relevant parts of the system are both reconstructable and predictable by the human operator, and take into account not only the information available to the user, but also limitations in how the user can process information about their own input.

C. Design constraints for the optimal output map

We wish to design user-interfaces that satisfy the following design constraints, in order of importance:

- 1) Availability of user-interface for the specified task
- 2) User trust of the automation
- 3) Minimization of information presented to the user

Constraint 1 ensures that the information presented to the user is sufficient for completing the task. Constraint 2 is important because it is often compromised in off-nominal scenarios, as the user may be unsure of the automation’s intent and actions. Constraint 3 guarantees conciseness, to help prevent high workload due to excessive information. Taken together, these three design constraints ensure that the user-interface has the appropriate amount of information to complete the specified task, depending on the user’s trust of the automation. We map these design constraints to an optimization problem in which the objective function arises from Constraint 3 and the optimization constraints arise from Constraints 1 and 2. Hence, we consider the following framework to construct an optimal user-interface:

$$\begin{aligned} &\text{minimize} && J(S) && \text{(information objective)} \\ &\text{subject to} && \begin{cases} S \in 2^S & \text{(possible outputs)} \\ S \in \mathcal{S}_{\text{avail}} \subseteq 2^S & \text{(availability constraint)} \\ S \in \mathcal{S}_{\text{trust}} \subseteq 2^S & \text{(trust constraint)} \end{cases} \end{aligned}$$

Given a task S_{task} , the objective is to identify the output map C_{S^*} associated with the optimal set of sensors S^* (from the possible set of choices 2^S) which meets all three design constraints for the MIMO system (1) or (2), respectively. We will define the sets $\mathcal{S}_{\text{avail}}$ and $\mathcal{S}_{\text{trust}}$ and the objective $J(\cdot)$ for each of the four scenarios considered, in Section III.

III. USER-INTERFACE DESIGN VIA SENSOR PLACEMENT

The most important design constraint that a user-interface must satisfy is that of availability. This constraint is common to all four scenarios we consider: nominal/off-nominal operation and noise-free/noisy measurements. For the system (8) with a controller that ensures $\xi(t)$ tracks $\xi_R(t)$, the method in [10] computes the user-observable and user-predictable subspaces, that is, it reconstructs both the current state and the current state derivative.

Lemma 1. *The external dynamics of the system (8) are user-observable, implying $\mathcal{X}_O = \mathbb{R}^{\gamma(S)}$.*

Proof. The external dynamics $\xi(t)$ associated with output map C_S , $S = \{s_1, s_2, \dots, s_{|S|}\} \in 2^S$ are comprised of the output channels $y_i(t) = s_i^\top x(t)$ and higher derivatives

$y_i^{(j)}(t)$, $1 \leq j \leq \gamma(s_i) - 1$, $1 \leq i \leq p$. By Assumption 1, $\xi(t)$ is user-observable, and $\mathcal{X}_O = \mathbb{R}^{\gamma(S)}$. ■

Lemma 2. *The user-predictable space associated with the external dynamics of (8) is $\mathcal{X}_P = T(S)\mathcal{N}((A_2 - B_m K_\eta(t))T^\perp(S))$ when the user-interface displays $\xi_c(t) = B_m K_\xi(t)\xi_R(t) + B_m u_{fw}(t)$.*

Proof. The term $\xi_c(t)$ in (8) is a function of the system parameters and $\xi_R(t)$. Hence, if the user-interface displays $\xi_c(t)$, then $\xi(t)$ can be estimated from the user-observable space whenever $(A_2 - B_m K_\eta(t))\eta(t) = 0$. When $\xi_c(t)$ is provided to the user, the user-predictable space is $\mathcal{X}_P = T(S)\mathcal{N}((A_2 - B_m K_\eta(t))T^\perp(S))$. ■

Hence, the task specified using S_{task} is feasible only if $S_{\text{task}} \in (T(S)\mathcal{N}((A_2 - B_m K_\eta(t))T^\perp(S))) \cap \mathbb{R}^{\gamma(S)}$ or $S_{\text{task}} \in T(S)\mathcal{N}((A_2 - B_m K_\eta(t))T^\perp(S))$. Therefore, a necessary condition for the task to be feasible is $S_{\text{task}} \in \mathcal{R}^\perp(T(S))$. Sufficient conditions would require more information regarding A_2 , B_m , and $K_\eta(t)$. If the choice of S results in $A_2 = 0$, then the zero dynamics and the external dynamics are decoupled in (5). This allows for the simplification of the LQR tracking problem, and the resulting system can be shown to have $\mathcal{X}_O = \mathcal{X}_P = \mathbb{R}^{\gamma(S)}$. Hence, under the hypothesis $A_2 = 0$, we translate the user-interface availability constraint as $S^* \in \mathcal{S}_{\text{avail}}$,

$$\mathcal{S}_{\text{avail}} = \{S \in 2^S : S_{\text{task}} \in \mathcal{R}^\perp(T(S))\}. \quad (9)$$

Next, we simplify $\mathcal{S}_{\text{avail}}$ for efficient computation. Let $T(S_{\text{task}})$ be the similarity transform that converts the MIMO system (1) or (2) to an input-output form when the output map is C_{task} . Clearly, $S_{\text{task}} \in \mathcal{R}^\perp(T(S_{\text{task}}))$. By definitions of $\mathcal{R}^\perp(\cdot)$ and $T(\cdot)$,

$$\mathcal{R}^\perp(T(S \cup S_{\text{task}})) = \mathcal{R}^\perp(T(S)) \cup \mathcal{R}^\perp(T(S_{\text{task}})). \quad (10)$$

Lemma 3. *If $S \in \mathcal{S}_{\text{avail}}$, then $\mathcal{R}^\perp(T(S_{\text{task}})) \subseteq \mathcal{R}^\perp(T(S))$ and $\gamma(S_{\text{task}}) \leq \gamma(S)$.*

Proof. If $S_{\text{task}} \in \mathcal{R}^\perp(T(S))$, then for $1 \leq i \leq |S_{\text{task}}|$, every row $c_i \in \mathbb{R}^{1 \times n}$ of C_{task} can be defined as $c_i = \sum_{j=1}^{\gamma(S)} \alpha_{ij} r_j$ where r_j are the rows of $T(S)$ and $\alpha_{ij} \in \mathbb{R}$ are the coefficients. By definition, for every i , $\mathcal{R}^\perp(T(S_{\text{task}}))$ covers the output channels $y_i(t) = c_i^\top x(t)$ and their higher derivatives till their corresponding relative degrees $\gamma(c_i)$ where these higher derivatives satisfy the condition $c_i A^k B = 0$ for $0 \leq k \leq \gamma(c_i) - 1$. Using the definition of c_i , $r_j A^k B = 0$ for $j \in I_{ij} = \{j \in [1, \gamma(S)] : \alpha_{ij} \neq 0\}$. This implies that $r_j A^k$ are the higher derivatives of the output channels of S which are uninfluenced by the input and are also spanned by $\mathcal{R}^\perp(T(S))$. Therefore, $\mathcal{R}^\perp(T(S_{\text{task}})) \subseteq \mathcal{R}^\perp(T(S))$. By definition of $\gamma(\cdot)$, $\gamma(S_{\text{task}}) \leq \gamma(S)$. ■

Proposition 1. $\mathcal{R}^\perp(T(S \cup S_{\text{task}})) = \mathcal{R}^\perp(T(S))$ if and only if $S \in \mathcal{S}_{\text{avail}}$.

Proof. ‘If’ statement: Straightforward using (10) and Lemma 3. ‘Only if’ statement: If $\mathcal{R}^\perp(T(S \cup S_{\text{task}})) = \mathcal{R}^\perp(T(S))$, we conclude $\mathcal{R}^\perp(T(S_{\text{task}})) \subseteq \mathcal{R}^\perp(T(S))$ using (10). Since $S_{\text{task}} \in \mathcal{R}^\perp(T(S_{\text{task}}))$, $S \in \mathcal{S}_{\text{avail}}$. ■

Lemma 4. $\mathcal{R}^\perp(T(S \cup S_{\text{task}})) = \mathcal{R}^\perp(T(S))$ if and only if $\gamma(S \cup S_{\text{task}}) = \gamma(S)$.

Proof. ‘If’ statement: Given $\gamma(S \cup S_{\text{task}}) = \gamma(S)$, assume for contradiction, $\mathcal{R}^\perp(T(S \cup S_{\text{task}})) \neq \mathcal{R}^\perp(T(S))$. Using (10), we know that $\mathcal{R}^\perp(T(S)) \subset \mathcal{R}^\perp(T(S \cup S_{\text{task}}))$ implying $\gamma(S) < \gamma(S \cup S_{\text{task}})$, a contradiction. ‘Only If’ statement: From definitions of $\gamma(\cdot)$ and $\mathcal{R}^\perp(T(\cdot))$. ■

From Proposition 1 and Lemma 4, we redefine (9) as

$$\mathcal{S}_{\text{avail}} = \{S \in 2^S : \gamma(S \cup S_{\text{task}}) = \gamma(S)\} \quad (11)$$

where $\gamma(\cdot)$ is computed using the Markov parameters. We note that $\mathcal{S}_{\text{avail}} \neq \{\emptyset\}$ since $S_{\text{task}} \in \mathcal{S}_{\text{avail}}$.

To facilitate computation of the optimal interface, we exploit the fact that $\gamma(S)$, the dimension of the external dynamics, is monotone increasing. Recall that a set function $\psi(S) : 2^S \rightarrow \{1, 2, 3, \dots, n\}$ is said to be monotone increasing if for all sets $S_1, S_2 \subseteq S$, $S_1 \subseteq S_2$ implies $\psi(S_1) \leq \psi(S_2)$ [23].

Proposition 2. $\gamma(S)$ is monotone increasing.

Proof. Follows from the definitions of $\gamma(\cdot)$ and $T(\cdot)$. ■

A. Nominal operation with noise-free measurements

For a task specified using S_{task} with $S_{\text{task}} \in 2^S$, Constraint 1 requires $\gamma(S) = \gamma(S \cup S_{\text{task}})$ from (11). To address Constraint 2, we exploit the fact that under nominal operating conditions, a trained user can rely on the automation to perform as expected. In other words, we presume that the user trusts the automation. We therefore aim to delegate authority to the automation, by reducing the cardinality of the user-observable subspace (and increasing the cardinality of the subspace that is unobservable to the user). Lastly, to address Constraint 3, we aim to construct an output map of minimal cardinality, hence

$$J(S) = |S|. \quad (12)$$

These three constraints are translated into a constrained optimization problem for the user-interface design for (8) in nominal operation with noise-free measurements:

Problem 1a (Nominal operation, noise-free measurement).

$$\begin{aligned} & \text{minimize} && |S| \\ & \text{subject to} && \begin{cases} S \in 2^S \\ \gamma(S) = \gamma(S \cup S_{\text{task}}) \\ \gamma(S) \leq \gamma(S'), \forall S' \in 2^S \end{cases} \end{aligned}$$

Problem 1a is a constrained combinatorial optimization problem. While $J(S)$ is a modular function, constrained optimization problems are hard to approximate using established algorithms when they do not have a special structure [31]. Therefore, we resort to computing the conditions for optimality and perform an exhaustive search in a smaller subset of 2^S . Let $\gamma_{\min} = \min_{S \in 2^S} \gamma(S)$, implying

$$\mathcal{S}_{\text{trust}} = \{S \in 2^S : \gamma(S) = \gamma_{\min}\} \quad (13)$$

By Proposition 2, computation of γ_{\min} requires a one-pass search in \mathcal{S} as opposed to 2^S . For Problem 1a, we can

constrain our search, based on (11) and (13), to the set of feasible solutions, defined as

$$\begin{aligned} \mathcal{S}_{\text{feas}} &= \mathcal{S}_{\text{avail}} \cap \mathcal{S}_{\text{trust}} \\ &= \{S \in 2^{\mathcal{S}} : \gamma(S) = \gamma(S \cup S_{\text{task}}) = \gamma_{\min}\}. \end{aligned} \quad (14)$$

Hence, the optimal solution S^* , if it exists, is an element of $\mathcal{S}_{\text{feas}}$ with the smallest cardinality.

Proposition 3. *If $\mathcal{S}_{\text{feas}}$ is not empty, then $S_{\text{task}} \in \mathcal{S}_{\text{feas}}$.*

Proof. If $\mathcal{S}_{\text{feas}} \neq \{\emptyset\}$, then $\exists S \in \mathcal{S}_{\text{feas}}$ such that $\gamma(S \cup S_{\text{task}}) = \gamma_{\min}$. By Proposition 2, $\gamma_{\min} \leq \gamma(S_{\text{task}}) \leq \gamma(S \cup S_{\text{task}}) = \gamma_{\min}$. Hence, $S_{\text{task}} \in \mathcal{S}_{\text{feas}}$. ■

Corollary 1. *Let $\mathcal{S}_{\text{feas}} \neq \{\emptyset\}$. An optimal solution S^* for Problem 1a is an element with the smallest cardinality of the set*

$$\mathcal{S}_{1a} = \mathcal{S}_{\text{feas}} \cap \{S \in 2^{\mathcal{S}} : |S| \leq |S_{\text{task}}|\}. \quad (15)$$

The set \mathcal{S}_{1a} is restricted to $S \in 2^{\mathcal{S}}$ such that $|S| \leq |S_{\text{task}}|$ because $S_{\text{task}} \in \mathcal{S}_{\text{feas}}$ whenever $\mathcal{S}_{\text{feas}} \neq \{\emptyset\}$ by Proposition 3. For a given system (8) and the set of sensors \mathcal{S} , the cardinality of \mathcal{S}_{1a} depends on S_{task} .

In the event that $\mathcal{S}_{\text{trust}} = \{\emptyset\}$ which implies $\mathcal{S}_{\text{feas}} = \{\emptyset\}$, we relax the constraint provided by Constraint 2 to $\gamma(S) \leq \gamma(S_{\text{task}})$ so that the set of feasible solutions is no longer empty. We define the relaxation of $\mathcal{S}_{\text{trust}}$ as

$$\mathcal{S}_{\text{trust}}^+ = \{S \in 2^{\mathcal{S}} : \gamma(S) \leq \gamma(S_{\text{task}})\} \neq \{\emptyset\}. \quad (16)$$

Problem 1b (Relaxation for Problem 1a).

$$\begin{aligned} &\text{minimize} && |S| \\ &\text{subject to} && \begin{cases} S \in 2^{\mathcal{S}} \\ \gamma(S) = \gamma(S \cup S_{\text{task}}) \\ \gamma(S) \leq \gamma(S_{\text{task}}) \end{cases} \end{aligned}$$

Since $S_{\text{task}} \in \mathcal{S}_{\text{avail}} \cap \mathcal{S}_{\text{trust}}^+$, an optimal solution always exists for Problem 1b. We define the relaxation of $\mathcal{S}_{\text{feas}}$ as

$$\mathcal{S}_{\text{feas}}^+ = \{S \in 2^{\mathcal{S}} : \gamma(S) = \gamma(S \cup S_{\text{task}}) = \gamma(S_{\text{task}})\}. \quad (17)$$

Lemma 5. $\mathcal{S}_{\text{feas}}^+ = \mathcal{S}_{\text{avail}} \cap \mathcal{S}_{\text{trust}}^+$.

Proof. By definition, $\mathcal{S}_{\text{feas}}^+ \subseteq (\mathcal{S}_{\text{avail}} \cap \mathcal{S}_{\text{trust}}^+)$. If $S \in \mathcal{S}_{\text{avail}} \cap \mathcal{S}_{\text{trust}}^+$, $\gamma(S \cup S_{\text{task}}) = \gamma(S)$ and $\gamma(S) \leq \gamma(S_{\text{task}})$ by (11) and (16). By Lemma 3, $\gamma(S_{\text{task}}) \leq \gamma(S) \Rightarrow \gamma(S_{\text{task}}) = \gamma(S)$. Hence, $S \in \mathcal{S}_{\text{feas}}^+$. ■

Proposition 4. *The optimal solution S^* for Problem 1b is the element with the smallest cardinality of the set*

$$\mathcal{S}_{1b} = \mathcal{S}_{\text{feas}}^+ \cap \{S \in 2^{\mathcal{S}} : |S| \leq |S_{\text{task}}|\}. \quad (18)$$

Corollary 1 and Proposition 4 substantiate the intuition that for a human-automation system in which the user trusts the automation completely, the user-interface should provide minimal information that is sufficient for the task at hand. We define the set in which the optimal solution is guaranteed to lie, \mathcal{S}_{nom} , as

$$\mathcal{S}_{\text{nom}} = \begin{cases} \mathcal{S}_{1a} & \mathcal{S}_{\text{feas}} \neq \{\emptyset\} \\ \mathcal{S}_{1b} & \text{otherwise} \end{cases}. \quad (19)$$

Algorithm 1 Nominal operation with noise-free measurements

Input: $S_{\text{task}}, \mathcal{S}_{\text{nom}}$ defined in (19)

Output: S^*

```

1: procedure NomOpt( $S_{\text{task}}, \mathcal{S}_{\text{nom}}$ )
2:   if  $|S_{\text{task}}|$  is 1 then
3:      $S^* \leftarrow S_{\text{task}}$ 
4:   else
5:      $S^* \leftarrow$  Smallest cardinality element in  $\mathcal{S}_{\text{nom}}$ 
6:   end if
7: end procedure

```

Irrespective of the cardinality of $\mathcal{S}_{\text{feas}}$, if $|S_{\text{task}}| = 1$, $S^* = S_{\text{task}}$ by Propositions 3 and 4. Since we want the minimum cardinality set $S^* \in \mathcal{S}_{\text{nom}}$, the exhaustive search in \mathcal{S}_{nom} can be done by searching in $2^{\mathcal{S}}$ in the increasing order of cardinality until a member of $2^{\mathcal{S}}$ is found which satisfies the constraints defining \mathcal{S}_{nom} . Note that $|\mathcal{S}_{\text{nom}}| \leq \sum_{i=1}^{|S_{\text{task}}|} \binom{|\mathcal{S}|}{i}$, a polynomial in $|\mathcal{S}|$ of the order $|S_{\text{task}}|$, which is much smaller than $2^{|\mathcal{S}|}$ as $|\mathcal{S}|$ increases.

As in Algorithm 1, the user-interface for nominal operation with noise-free measurements is

$$S^* = \text{NomOpt}(S_{\text{task}}, \mathcal{S}_{\text{nom}}). \quad (20)$$

B. Off-nominal operation with noise-free measurements

Off-nominal conditions occur in unusual (and often highly unlikely) situations, in which the user is often unfamiliar with the automation's behavior (e.g., the user's mental model may be inaccurate or highly uncertain). Additionally, the automation may not actually perform as intended (as documented in many aircraft incidents and accidents). Hence, in off-nominal scenarios, the user often has little trust in the automation, because the automation can appear to have unpredictable behavior. User-interfaces must satisfy vastly different design constraints in off-nominal scenarios.

In particular, the lack of trust in the automation leads to the need to *increase* the user-observable subspace. Unlike in nominal operation, in which responsibility is delegated to the automation, in off-nominal scenarios, the user should have more authority because the automation is not trusted. This is equivalent to maximizing the dimension of the external dynamics, or, equivalently, to reducing the dimension of the zero dynamics. Constraints 1 and 3 remain the same as in the nominal case: user-observability and user-predictability must be assured, and minimal cardinality of information in the user-interface is desirable.

For the MIMO system (8) with a task specified using S_{task} , the optimal output map for off-nominal operation scenarios is the solution to the following:

Problem 2a (Off-nominal operation, noise-free measure-

ments).

$$\begin{aligned} & \text{minimize} && |S| \\ & \text{subject to} && \begin{cases} S \in 2^S \\ \gamma(S) = \gamma(S \cup S_{\text{task}}) \\ \gamma(S) \geq \gamma(S'), \forall S' \in 2^S \end{cases} \end{aligned}$$

This problem can be recast as a set covering problem [32]. First, let the dimension of the largest user-observable space attainable be $\gamma_{\max} \triangleq \max_{S \in 2^S} \gamma(S)$. Using Proposition 2, we have $\gamma_{\max} = \gamma(S)$, which mitigates the need for performing any sort of search in 2^S . Hence

$$\mathcal{S}_{\text{trust}} = \{S \in 2^S : \gamma(S) = \gamma_{\max}\} \neq \{\emptyset\}. \quad (21)$$

All the states of (2) are observable for C_{S^*} when $\gamma_{\max} = n$.

Proposition 5. For $\mathcal{S}_{\text{trust}}$ in (21), $\mathcal{S}_{\text{trust}} \subseteq \mathcal{S}_{\text{avail}}$.

Proof. If $S \in \mathcal{S}_{\text{trust}}$, then $\mathcal{R}^\perp(T(S))$ is the largest observable space that can be constructed from S . Hence, for any $S_{\text{task}} \in 2^S$, $\mathcal{R}^\perp(T(S_{\text{task}})) \subseteq \mathcal{R}^\perp(T(S))$. By Proposition 1 and (10), $S \in \mathcal{S}_{\text{avail}}$. ■

Corollary 2. For off-nominal operation, user-interface design does not depend on the task $S_{\text{task}} \in 2^S$.

From Proposition 5, an optimal solution $S^* \in \mathcal{S}_{\text{trust}}$ always exists for Problem 2a. Corollary 2 arises since $\mathcal{S}_{\text{trust}}$ (21) is independent of S_{task} . We now define the set covering problem equivalent to Problem 2a using [32, Problem 2.1].

Problem 2b (Set covering problem equivalent to Problem 2a). From a collection of subspaces $\mathcal{R}^\perp(T(S)) \subseteq \mathbb{R}^{\gamma_{\max}}$ where $S \in 2^S$, find the minimum cardinality set S^* such that $\mathcal{R}^\perp(T(S^*)) = \mathbb{R}^{\gamma_{\max}}$.

The solution to Problem 2b is the minimal cardinality set of sensors which achieves the maximum observability possible amongst the set of feasible sensor combinations. Algorithm 2 is the best greedy algorithm to find a sub-optimal solution S^+ to Problem 2b [32].

Proposition 6 (Based on [32, Theorem 2.4]). If S^* is an optimal solution of Problem 2b, S^+ is such that $\gamma(S^+) = \gamma(S^*)$ and $|S^*| \leq |S^+| \leq |S^*| \sum_{k=1}^{|S^*|} \frac{1}{k}$.

Using Proposition 6, we define the set in which the optimal solution is guaranteed to lie, $\mathcal{S}_{\text{off-nom}}$, as

$$\begin{aligned} \mathcal{S}_{\text{off-nom}} &= \mathcal{S}_{\text{trust}} \cap \{S \in 2^S : |S| \leq |S^+|\} \\ &= \{S \in 2^S : \gamma(S) = \gamma_{\max}, |S| \leq |S^+|\}. \end{aligned} \quad (22)$$

Similar to the analysis of noise-free measurements under nominal operation, S^+ is optimal if $|S^+| = 1$. If $|S^+| > 1$, we must perform an exhaustive search in $\mathcal{S}_{\text{off-nom}}$. We complete the search in 2^S in increasing cardinality, until a solution is found which satisfies (22). Hence, using Algorithm 1,

$$S^* = \text{NomOpt}(S^+, \mathcal{S}_{\text{off-nom}}). \quad (23)$$

Note that $|\mathcal{S}_{\text{off-nom}}| \leq \sum_{i=1}^{|S^+|} \binom{|S|}{i}$, a polynomial in $|S|$ of the order $|S^+|$, which is much smaller than $2^{|S|}$ as $|S|$ increases. By Corollary 2, S^* is independent of S_{task} .

Algorithm 2 Greedy algorithm for sub-optimal solution S^+ for Problem 2b

Input: $S, \gamma(\cdot)$

Output: S^+

```

1: procedure OffGreedy( $\gamma(\cdot), S$ )
2:    $S^+ \leftarrow \{\emptyset\}$ 
3:    $s_{\max} \leftarrow \arg \max_{s \in S} \gamma(s)$ 
4:   do
5:      $S^+ \leftarrow S^+ \cup \{s_{\max}\}$ 
6:      $s_{\max} \leftarrow \arg \max_{s \in S \setminus S^+} (\gamma(S^+ \cup s) - \gamma(S^+))$ 
7:   while  $\gamma(S^+ \cup s_{\max}) > \gamma(S^+)$ 
8: end procedure

```

C. Nominal and off-nominal operation with noisy measurements

Noisy measurements (due to e.g., sensor noise or other errors) introduce additional complexity in the design of user-interfaces under both nominal and off-nominal scenarios. With noisy measurements, the information the user has access to is unreliable. One way to mitigate the effect of the uncertainty is to minimize the influence of the noise $v(t)$ on the observable state $\xi(t)$ in the system (1).

We presume the human to be a minimum mean-square error (MMSE) estimator, so our interpretation of Constraint 3 differs from the noise-free cases. Here, we aim to minimize the volume of the uncertainty ellipse $\mathcal{E}_{\text{err}} = \{e \in \mathbb{R}^n : e^\top \frac{W_S}{V^2} e \leq 1\}$ where W_S is the observability gramian associated with output matrix C_S and e is the error in estimation of $\xi(0)$. The volume of \mathcal{E}_{err} is

$$\text{Volume}(\mathcal{E}_{\text{err}}) = \frac{(\pi)^{\frac{\rho(W_S)}{2}} V^{\frac{2}{\rho(W_S)}}}{\Gamma(\frac{\rho(W_S)}{2} + 1)} \left(\prod_{i=0}^{\rho(W_S)} \lambda_i(W_S) \right)^{\frac{-1}{\rho(W_S)}}, \quad (24)$$

where $\lambda_i(W_S)$ are the non-zero eigenvalues of W_S [33].

For nominal operation, $\mathcal{S}_{\text{feas}}$ and $\mathcal{S}_{\text{feas}}^+$ are the sets of sensor combinations consistent with Constraints 1 and 2. For off-nominal operation, $\mathcal{S}_{\text{trust}}$ (21) is the set of sensors consistent with Constraints 1 and 2 by Proposition 5. Constraint 3 now captures minimization of uncertainty (instead of minimization of cardinality) in the information presented to the user. Inspired by [19], [22], [23], we define the objective function to be minimized as

$$J(S) = - \sum_{i=1}^{\rho(W_S)} \log(\lambda_i(W_S)) \quad (25)$$

The optimal output maps for nominal and off-nominal scenarios, respectively, for the system (1) with a task specified using S_{task} , are the solutions the following problems.

Problem 3a (Nominal operation, noisy measurement).

$$\begin{aligned} & \text{maximize} && \sum_{i=1}^{\rho(W_S)} \log(\lambda_i(W_S)) \\ & \text{subject to} && S \in \mathcal{S}_{\text{feas}} \end{aligned}$$

	Noise-free	Noisy
Nominal	Prob. 1a or 1b, based on $\mathcal{S}_{\text{feas}}$	Prob. 3a or 3b, based on $\mathcal{S}_{\text{feas}}$
	$S^* = \text{NomOpt}(S_{\text{task}}, \mathcal{S}_{\text{nom}})$ (Optimal: Algorithm 1)	Search in $\mathcal{S}_{\text{feas}}$ or $\mathcal{S}_{\text{feas}}^+$ (Optimal)
Off-Nominal	Prob. 2b	Prob. 4
	$S^+ = \text{OffGreedy}(\gamma(\cdot), \mathcal{S})$ (Sub-optimal: Algorithm 2)	Search in $\mathcal{S}_{\text{trust}}$ (Optimal)
$S^* = \text{NomOpt}(S^+, \mathcal{S}_{\text{off-nom}})$ (Optimal: Algorithm 1)		

TABLE I: Summary of Section III

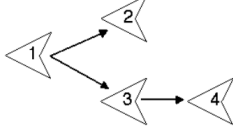


Fig. 1: Remotely controlled fleet of UAVs

Problem 3b (Relaxation for Problem 3a).

$$\begin{aligned} & \text{maximize} && \sum_{i=1}^{\rho(W_S)} \log(\lambda_i(W_S)) \\ & \text{subject to} && S \in \mathcal{S}_{\text{feas}}^+ \end{aligned}$$

Problem 4 (Off-nominal operation, noisy measurement).

$$\begin{aligned} & \text{maximize} && \sum_{i=1}^{\rho(W_S)} \log(\lambda_i(W_S)) \\ & \text{subject to} && S \in \mathcal{S}_{\text{trust}} = \{S \in 2^S : \gamma(S) = \gamma_{\text{max}}\} \end{aligned}$$

Note that $\mathcal{S}_{\text{trust}}$ and $\mathcal{S}_{\text{feas}}$ are equivalence classes with respect to relative degree, since $\gamma(\cdot)$ takes on the values γ_{max} and γ_{min} , respectively. Since $\rho(W_S) = \gamma(S)$, $\rho(W_S)$ is constant for all feasible S in Problems 3a and 4, $J(S)$ is submodular and monotone increasing. Unfortunately, the greedy heuristic for submodular maximization [22], [23] is applicable only to unconstrained optimization. The optimal solutions to Problems 3a, 3b, and 4 are found using exhaustive searches which may pose a higher computational burden as compared to the noise-free scenarios, depending on the values of $|\mathcal{S}_{\text{feas}}|$, $|\mathcal{S}_{\text{feas}}^+|$, and $|\mathcal{S}_{\text{trust}}|$, since we do not perform the search in 2^S in increasing cardinality.

Table I summarizes the results from all four scenarios.

IV. SIMULATIONS USING REMOTELY CONTROLLED FLEET OF UAVS

Consider a remotely controlled fleet of four homogeneous UAVs, as in [10]. The fleet has a leader-follower formation, with communication structure as shown in Figure 1. The UAVs move only in the horizontal direction and are individually modelled as double integrators. For the i^{th} UAV, the dynamics are given by $\dot{x}_v^i = A_v x_v^i + B_v u^i$, with UAV state $x_v^i \in \mathbb{R}^2$, input $u^i \in \mathbb{R}$, and matrices $A_v = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ and $B_v = [0 \ 1]^\top$ for $i \in \{1, 2, 3, 4\}$.

We define the state of the fleet as $x \in \mathbb{R}^8$. The human operator controls the leader, and the follower vehicles autonomously maintain a formation $h = h_p \otimes [1 \ 0]^\top$, with $h_p \in \mathbb{R}^4$ the desired position of the fleet [34]. We presume

a hierarchical control scheme, with an inner control loop to achieve the required formation h , and an outer loop to track the desired trajectory. As in [10], the fleet dynamics are

$$\begin{aligned} \dot{x}(t) = & I_4 \otimes A_v x(t) + \Gamma_L \otimes B_v F(x(t) - h) \\ & + \Gamma_F \otimes B_v z(t) \end{aligned} \quad (26)$$

where $F = [-1 \ -4]$ and $z(t) \in \mathbb{R} \times \{0\}^3$ is the external control input which affects only the leader. The gain F is chosen such that $A_v - B_v F$ is Hurwitz [34]. The matrices Γ_L, Γ_F are

$$\Gamma_L = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}, \quad \Gamma_F = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

We transform the state to $\bar{x}(t) = x(t) - h$, and write the fleet dynamics (with measurement noise)

$$\begin{aligned} \dot{\bar{x}}(t) &= \bar{A} \bar{x}(t) + \bar{B} z(t) \\ \bar{y}(t) &= \bar{C} \bar{x}(t) + v(t) \end{aligned} \quad (27)$$

with matrices $\bar{A} = I_4 \otimes A_v + \Gamma_L \otimes B_v F$, $\bar{B} = \Gamma_F \otimes B_v$, and with $v(t) \in L_2([0, \infty))$ where $\|v(t)\|_2 \leq V^2$.

We define the set of possible outputs to be the position and speed of each vehicle, $\mathcal{S} = \{s_1, s_2, \dots, s_8\}$, where $s_i \in \mathbb{R}^{1 \times 8}$ and $s_i = \mathbf{e}_i$, the unit vector in direction i . We wish to construct the optimal user-interface \bar{C} for both nominal and off-nominal operation, in the presence and absence of measurement noise. Consider two control objectives of moving the fleet: 1) *waypoint tracking* in which the leader moves to a specific waypoint (or set of waypoints), and 2) *trajectory tracking* in which the leader tracks a time-varying profile of position and speed. We define the tasks for waypoint tracking and trajectory tracking using $S_{\text{waypoint}} = \{s_1\}$ and $S_{\text{trajectory}} = \{s_1, s_2\}$ respectively.

Constraint 1 ensures availability of a user-interface for the specified task — for waypoint tracking, $\gamma(S) = \gamma(S \cup S_{\text{waypoint}})$ and for trajectory tracking, $\gamma(S) = \gamma(S \cup S_{\text{trajectory}})$. Constraint 2 under nominal operation (e.g., no unusual disruptions, disturbances, or other unexpected phenomena) requires the user-interface to have the least possible dimension of the user-observable space, γ_{min} . Under off-nominal operation (e.g., the fleet flies close to a restricted area and triggers envelope protection schemes, or other control algorithms that are unfamiliar to the operator), Constraint 2 requires the largest possible dimension of the user-observable space, γ_{max} . In off-nominal operation, Constraint 2 is sufficient for Constraint 1 by Corollary 2. Using (13) for nominal operation, $\mathcal{S}_{\text{trust}} = \{s_2\}$. Since $s_2 \notin \mathcal{S}_{\text{avail}}$, $\mathcal{S}_{\text{feas}} = \{\emptyset\}$ by (14). Hence, we compute $\mathcal{S}_{\text{feas}}^+ = \{\{s_1\}, \{s_1, s_2\}\}$ using (17) for both S_{waypoint} and $S_{\text{trajectory}}$. Constraint 3 is captured in the cost function (12) or (25) for noise-free or noisy measurements, respectively.

1) *Noise-free case*: For the nominal operation, the zero dynamics are decoupled from the external dynamics for every $S \in \mathcal{S}_{\text{feas}}^+$, which means that the user-interface availability is equivalently captured by $\gamma(S) = \gamma(S \cup S_{\text{task}})$. For

both waypoint and trajectory tracking, $S^* = \{s_1\}$ from Algorithm 1. Knowledge of the position of the leader UAV is sufficient when the user trusts the automation.

For off-nominal operation, since $\gamma(\mathcal{S}) = \gamma_{\max}$, the optimal output map will make the system user-observable. From Algorithm 2, we obtain $S^+ = \{s_3, s_7\}$ and $\gamma(S^+) = 8$. Since $\forall s \in \mathcal{S}$, $\gamma(s) < 8$, we have $|S^*| > 1$, and by Proposition 6, we conclude $S^* = S^+$ for both tasks. Since the user may not trust the automation, the user will need to estimate the position of all the UAVs.

2) *Noisy case*: For nominal operation, $S^* = \{s_1, s_2\}$ solves Problem 3b. Thus, under nominal operation, the velocity of the leader UAV is required (unlike nominal operation in the noise-free case).

For off-nominal operation, S_{trust} consists of all sensor combinations that contain the position or speed of each of vehicles 2 and 4 (the vehicles furthest from the leader in the communication structure). We select the combination with the fewest elements and least differentiations of the output $y(t)$, that is, $S^* = \{s_4, s_8\}$ (speeds of vehicles 2 and 4).

V. CONCLUSION AND FUTURE WORK

This paper identifies design constraints which translate human factors guidelines for good human-automation interaction in manual control systems, to sensor placement problems. The user-interface design problem is posed as a combinational optimization. We construct optimal output maps for nominal and off-nominal operation of a MIMO LTI system, with and without noisy measurements. These output maps enable synthesis of user-interfaces which will satisfy necessary (and in some cases, sufficient) conditions for effective operation. Sub-optimal solutions were obtained for these problems, and the necessary conditions for the optimal solutions to these problems were characterized.

Future work includes the development of more efficient algorithms, extensions to hybrid LTI systems, and to multi-modal user-interfaces.

REFERENCES

- [1] A. J. Dix, *Formal Methods for Interactive Systems*. London, England: Academic Press, 1991.
- [2] C. Billings, *Aviation Automation: The Search for a Human-Centered Approach*. Hillsdale, NJ: Erlbaum, 1997.
- [3] J. Bailleul, N. E. Leonard, and K. Morgansen, "Interaction dynamics: The interface of humans and smart machines," *Proceedings of the IEEE*, vol. 100, no. 3, pp. 567–570, 2012.
- [4] G. Gelman, K. Feigh, and J. Rushby, "Example of a complementary use of model checking and human performance simulation," *IEEE Trans. Human-Machine Systems*, vol. 44, no. 5, pp. 576–590, October 2014.
- [5] A. Pritchett and M. Feary, "Designing human-automation interaction," in *Handbook of Human-Machine Interaction*, G. Boy, Ed. Ashgate, 2011, pp. 267–282.
- [6] M. L. Bolton, E. Bass, and R. Siminiceanu, "Using formal verification to evaluate human-automation interaction: A review," *IEEE Trans. Systems, Man, and Cybernetics: Systems*, vol. 43, no. 3, pp. 488–503, 2013.
- [7] A. Degani and M. Heymann, "Formal verification of human-automation interaction," *Human Factors*, vol. 44, no. 1, pp. 28–43, 2002.
- [8] M. Oishi, I. Mitchell, A. Bayen, A. Degani, and C. Tomlin, "Invariance-preserving abstractions of hybrid systems: Application to user interface design," *IEEE Trans. on Control Systems Technology*, vol. 16, no. 2, pp. 229–244, March 2008.
- [9] N. Eskandari and M. Oishi, "Computing observable and predictable subspaces to evaluate user-interfaces of LTI systems under shared control," in *IEEE Conference on Systems, Man and Cybernetics*, Alaska, USA, October 2011, pp. 2803–2808.
- [10] M. M. K. Oishi, "Assessing information availability for user-interfaces of shared control systems under reference tracking," in *American Control Conference*, 2014, pp. 3474–3481.
- [11] T. Hammond, N. Eskandari, and M. Oishi, "Observability of user-interfaces for hybrid LTI systems under collaborative control: Application to aircraft flight management systems," *IEEE Trans. Automation Science and Engineering*, vol. 13, no. 1, pp. 78–84, 2015.
- [12] M. P. Vitus and C. J. Tomlin, "Sensor placement for improved robotic navigation," in *Proceedings of Robotics: Science and Systems*, Zaragoza, Spain, June 2010.
- [13] A. Mourikis, S. Roumeliotis *et al.*, "Optimal sensor scheduling for resource-constrained localization of mobile robot formations," *IEEE Trans. Robotics*, vol. 22, no. 5, pp. 917–931, 2006.
- [14] H. Rowaihy, S. Eswaran, M. Johnson, D. Verma, A. Bar-Noy, T. Brown, and T. La Porta, "A survey of sensor selection schemes in wireless sensor networks," in *Defense and Security Symposium*, 2007.
- [15] J. Qi, K. Sun, and W. Kang, "Optimal PMU placement for power system dynamic state estimation by using empirical observability gramian," *IEEE Trans. Power Systems*, vol. 30, pp. 2041–2054, 2015.
- [16] B. Gou, "Generalized integer linear programming formulation for optimal PMU placement," *IEEE Trans. Power Systems*, vol. 23, no. 3, pp. 1099–1104, 2008.
- [17] M. Van De Wal and B. De Jager, "A review of methods for input/output selection," *Automatica*, vol. 37, no. 4, pp. 487–510, 2001.
- [18] J. L. Crassidis and J. L. Junkins, *Optimal estimation of dynamic systems*. CRC press, 2011.
- [19] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. Signal Processing*, vol. 57, no. 2, pp. 451–462, 2009.
- [20] B. Polyak, M. Khlebnikov, and P. Shcherbakov, "An LMI approach to structured sparse feedback design in linear control systems," in *European Control Conference*, 2013, pp. 833–838.
- [21] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *AAAI*, vol. 7, 2007, pp. 1650–1654.
- [22] M. Shamaiah, S. Banerjee, and H. Vikalo, "Greedy sensor selection: Leveraging submodularity," in *IEEE Conference on Decision and Control*, 2010, pp. 2572–2577.
- [23] T. H. Summers, F. L. Cortesi, and J. Lygeros, "On submodularity and controllability in complex dynamical networks," *IEEE Trans. Control of Network Systems*, vol. 3, no. 1, pp. 1–11, 2016.
- [24] T. Y. Berger-Wolf, W. E. Hart, and J. Saia, "Discrete sensor placement problems in distribution networks," *Mathematical and Computer Modelling*, vol. 42, no. 13, pp. 1385–1396, 2005.
- [25] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions - I," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.
- [26] M. Endsley, *Designing for Situation Awareness: An Approach to User-Centered Design*, 2nd ed. CRC Press, 2011.
- [27] C. D. Wickens, "Situation Awareness: Review of Mica Endsley's 1995 articles on Situation Awareness Theory and Measurement," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 50, no. 3, pp. 397–403, 2008.
- [28] A. Isidori, *Nonlinear Control Systems*, ser. Communications and Control Engineering. Springer London, 1995.
- [29] S. Sastry, *Nonlinear Systems: Analysis, Stability, and Control*. Springer New York, 2013.
- [30] P. Dorato, V. Cerone, and C. Abdallah, *Linear-quadratic control: An Introduction*. Simon & Schuster, 1994.
- [31] S. T. McCormick, "Submodular function minimization," *Handbooks in operations research and management science*, vol. 12, pp. 321–391, 2005.
- [32] V. V. Vazirani, *Approximation Algorithms*. Springer Science & Business Media, 2013.
- [33] G. E. Dullerud and F. Paganini, *A Course in Robust Control Theory: A Convex Approach*. Springer Verlag, 2000.
- [34] G. Lafferriere, A. Williams, J. Caughman, and J. Veerman, "Decentralized control of vehicle formations," *Systems & Control Letters*, vol. 54, no. 9, pp. 899–910, 2005.