Semi-Autonomous Systems with Contextual Competence Awareness

Saaduddin Mahmud*
University of Massachusetts Amherst
Amherst, USA
smahmud@cs.umass.edu

Connor Basich*
University of Massachusetts Amherst
Amherst, USA
cbasich@cs.umass.edu

Shlomo Zilberstein
University of Massachusetts Amherst
Amherst, USA
shlomo@cs.umass.edu

ABSTRACT

Competence modeling is critical for the efficient and safe operation of semi-autonomous systems (SAS) with varying levels of autonomy. In this paper, we extend the notion of competence modeling by introducing a *contextual competence* model. While previous work on competence-aware systems (CAS) defined the competence of a SAS relative to a single static operator, we present an augmented operator model that is contextualized by Markovian state information capable of capturing multiple operators. Access to such information allows the SAS to account for the stochastic shifts that may occur in the behavior of the operator(s) during deployment and optimize its autonomy accordingly. We show that the extended model called Contextual Competence Aware System (CoCAS) has the same convergence guarantees as CAS, and empirically illustrate the benefit of our approach over both the original CAS model as well as other relevant work in shared autonomy.

KEYWORDS

Semi-Autonomous Systems; Markov Decision Processes; Adjustable Autonomy; Competence-Aware Systems

ACM Reference Format:
Saaduddin Mahmud*, Connor Basich*, and Shlomo Zilberstein. 2023. Semi-Autonomous Systems with Contextual Competence Awareness. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2 2023*, IFAAMAS, 9 pages.

1 INTRODUCTION

Recent efforts towards the deployment of autonomous systems in the open world have spurred an increased interest in the development of systems that strategically manage the trade-off between fully autonomous operation and partial reliance on human assistance. Such systems are often desirable due to (1) their ability to more effectively handle a wider array of tasks than either agent could execute on their own [9, 41], and (2) their ability to ensure safe operation and risk mitigation [16, 20]. These properties are particularly desirable in many open-world domains such as autonomous driving [7, 8, 30, 36], medical robotics [15, 40], and extraterrestrial science robotics [17, 23, 34].

In this paper, we focus on *semi-autonomous systems* [41] that can operate autonomously under certain conditions, and may otherwise require human assistance to achieve their assigned task. Specifically, the aim of this paper is to extend the notion of *competence modeling* we introduced in prior work [2] where we define competence in the context of discrete *levels of autonomy* [24]. Each level of autonomy gives the human operator a specific assistive role, and the overall goal is to optimize the system's reliance on the human operator. In other words, the goal is to optimize the trade-off between autonomous operation and human assistance, given the associated costs and benefits of the available forms of human help.

As discussed by Costen et al. [12], the *competence-aware system* (CAS) framework introduced in [2] relies on a number of strong assumptions about "human authority" that may diminish the applicability of the framework to certain real-world domains. In particular, the original model assumes that (1) there is only a single human operator, (2) the human is perfectly consistent, and (3) the human is perfectly safe and has no variation in performance. Essentially the human is treated as a stationary black box oracle. However, humans are not perfectly consistent and may themselves have variations in performance due to differences in skills or state. This is particularly relevant as the proposed notion of competence is specifically a measure of an autonomous agent's ability to operate in a shared-autonomy setting rather than its underlying technical capacity directly, and is hence fundamentally conditioned on its capacity to interact with the human operator.

On the other hand, Costen et al. [12] focuses specifically on the shared-autonomy problem of optimal operator selection in a stochastic multi-operator context where the current state of each operator is never directly observable. However, the proposed *stochastic-operator semi-autonomous system* (SO-SAS) model does not consider several features we believe are highly relevant in many current shared-autonomy settings. First, the authors do not consider multiple levels of autonomy, proposing an all-or-nothing operative control structure; second, each operator behaves according to a fixed policy when in control; and third, the reward function is independent of the operator (up to the transition dynamics).

In this work, we are primarily interested in how the concepts of both competence modeling and stochastic operator models can be applied more generally to autonomous systems. Developing such a model is important in the context of a fleet of agents that are, collectively, supported by a (smaller) team of remote (human) teleoperators. We believe that such systems will become increasingly prevalent as semi-autonomous systems are deployed in greater numbers in more complicated domains, necessitating human involvement for either technical, ethical, or legal reasons [19, 21, 22].

Example: Consider a fleet of semi-autonomous vehicles used to deliver packages that are supported by a small group of remote human

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2 2023, London, United Kingdom.

* These authors contributed equally to this work.

operators. The vehicles may require varying levels of tele-operative support to handle challenging, hazardous, or socially ambiguous situations that may constrain the autonomous capacity of the vehicles. For instance, tele-operative control of the vehicle may be required to complete a challenging maneuver, or to approve dropping off a package when no human is around to accept it. However, the availability and capabilities of the tele-operators may vary with time and global contextual information such as the current demand, the weather, or the time of day.

We propose a natural extension of our previous framework, competence-aware systems Basich et al. [2], which includes multiple stochastic human operators in a capacity similar to what was proposed by Costen et al. [12]. The model, which we call *contextual competence-aware systems* (CoCAS), can operate in multiple levels of autonomy and is capable of optimizing its autonomy with respect to global contextual information about the world, and local contextual information about each human operator. We show that not only can the CoCAS model fully represent the CAS model as defined in [2], but also the stochastic-operator SAS (SO-SAS) model defined in [12] when the state of each operator is fully observable.

Importantly, our model addresses the limitations of both models without sacrificing the representational power of either. Whereas the SO-SAS model only considers operators (autonomous and human) with predefined fixed policies and separates the decision-making of each entity from the selection of the entity, leading to potentially suboptimal behavior by the autonomous operator, the CoCAS model integrates both levels of decision-making together, leading to more proactive and globally robust decision-making that improves the overall performance of the system.

2 RELATED WORK

There is a growing body of research on when and how an autonomous system should proactively seek human assistance. Wray et al. [35] formalized the decision-theoretic planning problem for a semi-autonomous system of type II (SAS-II) with multiple actors and made explicit the problem of *transferring control* between actors in a SAS-II, focusing on uncertainty surrounding the capacity of the human driver to take control of the system at any given time in a safe capacity [35, 37, 38]. Costen et al. [12] extended this framework to consider *stochastic operators* whose internal states, and consequently the performance of each operator, can vary through time on more impacting factors than just their ability to arrest control of the system. The authors focus on the problem of determining which among a number of operators for a task (one of which is the autonomous robotic agent) should be given control at any given action-step in order to maximize the utility over the task's entire horizon conditioned on the agent's belief over the state of each operator but assume a fixed policy pre-computed for each operator, separating the "second-to-second" decision-making of the execution of the task from who is controlling that execution.

In prior work, we also extended the semi-autonomous systems framework by integrating a notion of *competence modeling* predicated on discrete *levels of autonomy* in what we call a *competence-aware system* (CAS) [2]. Levels of autonomy is a paradigm for reasoning about the trade-off in autonomous operation and human assistance, that is, gradations in limitations in autonomous operation and their commensurate levels and forms of human assistance [24], and has been well explored in both the academic literature [4, 11, 24, 31] as well as industrial applications [3, 29, 39]. In particular, a CAS learns its competence from human feedback acquired through various forms of interaction and integrates the learned competence model into its planning process. However, the CAS framework is only considered in the setting where there is a single stationary human authority and does not account for the possibility of multiple heterogeneous humans, or non-stationary human states.

Several other forms of shared autonomy (SA) that vary the dynamics and capacity of how autonomy is shared between multiple actors have been proposed over the years. Mixed-initiative control is a shared autonomy framework in which a human operator and an autonomous agent may individually take initiative to operate at different stages of execution to best leverage the utilities of the respective actors [10, 11, 14, 18]. More related to the ideas discussed in this paper comes from Petousakis et al. [25] who furthered Chiou et al. [11]'s approach by modeling the *cognitive availability* of the human expert based on real-time vision of the human to better inform the control-switching decision. Our approach differs from this area of research in that we assume that an automated planner determines the optimal level of autonomy for an agent to carry out rather than letting each entity initiate control independently.

Symbiotic autonomy [6, 28, 32, 33] is another related area of shared autonomy where the human and agent are separated as entities operating in the same environment (that is, operating independently in pursuit of potentially independent goals). In symbiotic autonomy, each agent may benefit from helping the other to achieve the others' goals in a symbiotic capacity, but their objectives may be independent. This differs from the setting that we consider where there is only a single objective and a single decision-making agent that can utilize human operators as a resource to provide assistance as needed in order to best achieve the objective.

3 BACKGROUND

In this section, we briefly describe the relevant background material needed to understand the contributions proposed in this paper. The competence-aware system (CAS) [2] is a planning model defined by integrating three underlying models: the (1) *domain model*, (2) *autonomy model*, and (3) *human feedback model*. The domain model is originally defined as a stochastic shortest path problem (SSP) [5], although in our work we instead model it more generally as a Markov decision process (MDP) [26].

An MDP is represented by the tuple $\langle S, A, T, R, \gamma \rangle$ where

- S is a finite set of states,
- A is a finite set of actions,
- $T : S \times A \times S \to [0, 1]$ is a transition function representing the probability of arriving in state s' having taken the action a in state s,
- $R : S \times A \to \mathbb{R}$ is a reward function representing the immediate expected reward of taking action a in state s, and
- $\gamma \in (0, 1]$ is the discount factor.

A solution to an MDP is a *policy*, denoted $\pi : S \to A$, which maps states to actions. A policy π induces the state-value function $V^\pi : S \to \mathbb{R}$, defined as $V^\pi(s) = R(s, \pi(s)) + \sum_{s' \in S} T(s, \pi(s), s')V^\pi(s')$, which represents the expected cumulative reward when starting in the state s and following the policy π. Similarly, a policy π induces the action-value function $q^\pi : S \times A \rightarrow \mathbb{R}$, defined as $q^\pi(s,a) = R(s,a) + \sum_{s' \in S} T(s,a,s')V^\pi(s')$ representing the expected cumulative reward when starting in state s, taking action a, and then following policy π.

The autonomy model is represented by the tuple $\langle \mathcal{L}, \kappa, \mu \rangle$ where, $\mathcal{L} = \{l_0, ..., l_n\}$ is the finite, partially ordered set of levels of autonomy where each l_i corresponds to some set of constraints on the system's autonomous operation; $\kappa : S \times A \rightarrow \mathcal{P}(\mathcal{L})$ is the *autonomy profile* mapping states $s \in S$ and actions $a \in A$ to a subset of $\mathcal{L}$ (note that $\mathcal{P}(\mathcal{L})$ denotes the powerset of $\mathcal{L}$), prescribing constraints on the allowed levels of autonomy for any situation; and $\mu : S \times \mathcal{L} \times A \times \mathcal{L} \rightarrow \mathbb{R}$ represents the *cost of autonomy* of performing action $a \in A$ at level $l' \in \mathcal{L}$ given that the agent is in state $s \in S$ and just operated in level $l \in \mathcal{L}$ in the previous state. Finally, the human feedback model describes the agent's knowledge about, and predictions of, its interactions with the human. Notably, this human feedback model only considers a single human authority and does not use any available contextual information about the human operator to reason about stochastic changes in human behavior, state, or capability.

4 CONTEXTUAL COMPETENCE-AWARE SYSTEMS

In this section, we outline and define the Contextual CAS (CoCAS) model. The CoCAS model includes one autonomous agent who is the primary operative agent, and $N(\geq 1)$ human operators $\mathcal{H} = \{H_1, ..., H_N\}$. Each human operator H_i has an operator state, θ_i, from a set of possible human states, $\Theta_{\mathcal{H}_i}$, at each timestep. Each $\Theta_{\mathcal{H}_i}$ may be unique but we henceforth simply write $\Theta_{\mathcal{H}}$ for notational convenience by observing that we could set a single $\Theta_{\mathcal{H}} = \cup_{i=1:N}\Theta_{\mathcal{H}_i}$. Additionally, the operator model may encode additional context information $\theta_c \in \Theta_C$. In general, $\Theta_{\mathcal{H}}$ encodes local information about an individual operator, such as whether they are tired or awake, their current capabilities as a tele-operator or authority, or the quality of their connection to the autonomous agent. On the other hand, Θ_C represents global information about the world, the set of operators as a whole, and other external contextual elements that may impact the competence of the system, for instance, the current active demand by other agents in the fleet, which operators are currently busy or available, the global weather conditions, etc.

In this work, we employ a more general domain model than in the original CAS formulation [2], represented as an arbitrary reward MDP. The autonomy model remains unchanged from the CAS model. Our main extension of CAS lies in how we extend the human feedback model to account for the stochastic change in the behavior of the human operators. Consequently, we refer to the new model used as the *operator model* due to the addition of contextual information regarding the operators that lies beyond the scope of just feedback. Note that unlike Costen et al. [12] we consider full observability of the contextual information in the operator model. In the context of tele-operating a fleet of semi-autonomous agents with a team of human operators; the contextual information discussed above can be reasonably acquired and used to optimize autonomy. We now describe our proposed operator model.

4.1 Operator Model

We formally represent the operator model as the tuple $\langle S_{\mathcal{H}}, T_{\mathcal{H}}, \Sigma, \lambda, \rho, \tau_{\mathcal{H}} \rangle$ where

- $S_{\mathcal{H}} = \Theta^N_{\mathcal{H}} \times \Theta_C$ represents the joint operator state of N the operators with additional information. It consists of N+1 factors. The i^{th} factor represents the *state* of the i^{th} operator (e.g., busy, active, tired, etc.). The $N+1^{th}$ factor indicates additional contextual information (e.g., current active operator, network quality, etc.),
- $T_{\mathcal{H}} : S_{\mathcal{H}} \times S \times \mathcal{L} \times A \times \mathcal{L} \rightarrow \Delta^{|S_{\mathcal{H}}|}$ is the *operator-state transition function* and represents how the operator-state vector, $S_{\mathcal{H}}$, may change conditioned on the agent taking action a at level l in state s having just acted in level l', given operator state $s_{\mathcal{H}}$,
- $\Sigma = \{\sigma_1, ..., \sigma_k\}$ is the set of possible feedback signals the agent can receive from each operator,
- $\lambda : S_{\mathcal{H}} \times S \times \mathcal{L} \times A \times \mathcal{L} \rightarrow \Delta^{|\Sigma|}$ is the *feedback profile* that represents the probability of receiving signal σ from the current operator i when performing action $a \in A$ at level $l' \in \mathcal{L}$ given that the agent is in state $s \in S$ and just operated in level $l \in \mathcal{L}$, and conditioned on the i^{th} operator's state,
- $\rho : S_{\mathcal{H}} \times S \times \mathcal{L} \times A \times \mathcal{L} \rightarrow \mathbb{R}^+$ is the *human cost function* and represents the positive cost to the current operator i of performing action $a \in A$ at level $l' \in \mathcal{L}$ given that the agent is in state $s \in S$ and just operated in level $l \in \mathcal{L}$, and conditioned on the i^{th} operator's state, an
- $\tau_{\mathcal{H}} : S_{\mathcal{H}} \times S \times A \rightarrow \Delta^{|S|}$ is the *human state transition function* that represents the probability of the current operator i taking the agent to state $s' \in S$ when the agent attempted to perform action $a \in A$ in state $s \in S$ but the operator took over control, conditioned on the i^{th} operator's state,

Notably, unlike in [12] where the transition model of each operator is dependent only on the state of each operator, above transition model can be affected by both the agents state, action, and level of autonomy. We believe that this better reflects domains in the real world where the operators may themselves be responsive to the operation of the agent. For example, if the agent moves into a hazardous area or attempts to perform a safety-critical or high-value action, one or more operators may be notified that urgent attention is needed prompting a change in operator state.

4.2 CoCAS Model

A *contextual competence-aware system* (CoCAS) is the natural extension to the CAS model with the new operator model defined above. Unlike a standard CAS model, the CoCAS accounts for the operator in control, and their state, when computing its competence and the best level of autonomy to act in. Additionally, by integrating $T_{\mathcal{H}}$, the transition model over operator states, into the planning model, a CoCAS can proactively plan in ways that optimize (in expectation) the likelihood of interacting with the right operator in the right state when needed.

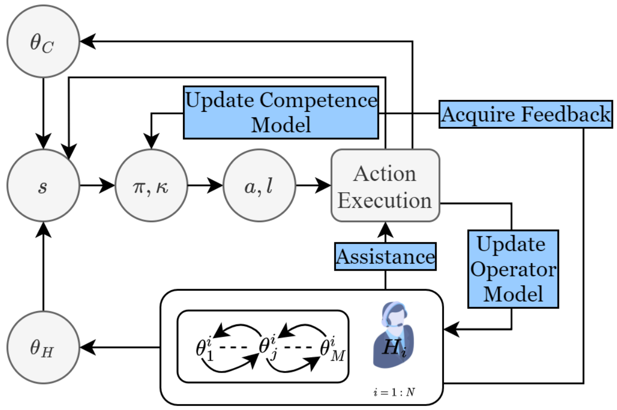


Figure 1: Illustration of the information flow in a CoCAS.

Definition 1. A **contextual competence-aware system** $\mathcal{S}$ is represented by the tuple $\langle \overline{S}, \overline{A}, \overline{T}, \overline{R}, \gamma \rangle$, where

- $\overline{S} = S_{\mathcal{H}} \times S \times \mathcal{L}$ is a set of factored states such that S is the set of domain states and $\mathcal{L}$ is the levels of autonomy,
- $\overline{A} = A \times \mathcal{L}$ is a set of factored actions such that A is the set of domain actions and $\mathcal{L}$ is the levels of autonomy,
- $\overline{T} : \overline{S} \times \overline{A} \to \Delta^{|\overline{S}|}$ is a transition function comprised of a state transition function $T_l : \overline{S} \times A \to \Delta^{|\overline{S}|}$ for each level $l \in \mathcal{L}$, and $T_{\mathcal{H}}$,
- $\overline{R} : \overline{S} \times \overline{A} \to \mathbb{R}$ is a reward function comprised of $R : S \times A \to \mathbb{R}$, $\mu : \overline{S} \times \overline{A} \to \mathbb{R}$, and $\rho : \overline{S} \times \overline{A} \to \mathbb{R}$, and
- $\gamma \in (0, 1]$ is the discount factor.

While construction of $\overline{S}$ and $\overline{A}$ is intuitive from the definition above, consider the following example of construction of $\overline{T}$ based on the example provided in [2], which we repeat here for clarity: Let $\mathcal{L} = \{l_0, l_1, l_2, l_3\}$ correspond to No Autonomy, Verified Autonomy (taking a action requires human approval), Supervised Autonomy (human operator is their to take over control if needed), and Full Autonomy. Also let $\Sigma = \{\emptyset, \oplus, \ominus, \oslash\}$, corresponding to *no feedback*, *approval*, *disapproval*, and *override* respectively. Furthermore, we assume that $\oplus$ and $\ominus$ can only be received in l_1, and $\oslash$ and $\emptyset$ only in l_2.We can now specify the **state transition function** of this CoCAS. Given $\bar{s}, \bar{a}$, and $\bar{s}'$, we define $\overline{T}$ as follows:

$$\overline{T}(\bar{s}, \bar{a}, \bar{s}') = \begin{cases} \tau_{\mathcal{H}}((s, s_{\mathcal{H}}), a, s') * \mathbf{T}_{\mathcal{H}}(\bar{\mathbf{s}}, \bar{\mathbf{a}}, \mathbf{s}'_{\mathcal{H}}), & \text{if } l = l_0, \\ \Big(\lambda(\oplus|\bar{s}, \bar{a}) T(s, a, s') + \\ \lambda(\ominus)[s = s']\Big) * \mathbf{T}_{\mathcal{H}}(\bar{\mathbf{s}}, \bar{\mathbf{a}}, \mathbf{s}'_{\mathcal{H}}), & \text{if } l = l_1, \\ \Big(\lambda(\emptyset|\bar{s}, \bar{a}) T(s, a, s') + \\ \lambda(\oslash|\bar{s}, \bar{a}) \tau_{\mathcal{H}}(s, a, s')\Big) * \mathbf{T}_{\mathcal{H}}(\bar{\mathbf{s}}, \bar{\mathbf{a}}, \mathbf{s}'_{\mathcal{H}}), & \text{if } l = l_2, \\ T(s, a, s') * \mathbf{T}_{\mathcal{H}}(\bar{\mathbf{s}}, \bar{\mathbf{a}}, \mathbf{s}'_{\mathcal{H}}), & \text{if } l = l_3, \end{cases}$$

where $\lambda(\cdot) = \lambda(\cdot|\bar{s}, \bar{a})$ and $[\cdot]$ denotes Iverson brackets, and bolded text indicates CoCAS-specific elements of the transition dynamics.

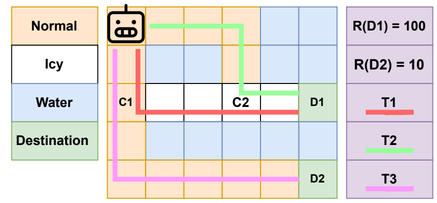


Figure 2: A simple MDP to explain CoCAS, CAS, and SO-SAS

We now present a simple example scenario to demonstrate the difference in the policy calculated under different shared autonomy models (CoCAS, CAS, and SO-SAS).

Example 1. (Fig 2): Consider the scenario where the robot has to navigate from its current location to one of the destinations (D1 and D2). The reward for $D1$ is greater. Further, going into the water is prohibited. We also have two types of human operators; experts and apprentices. The robot and the apprentices are only capable of navigating in the normal cells and if they move on icy cells they will fall into nearby water cells with a high probability. On the other hand, the expert can navigate in both normal and icy cells. At the beginning of each episode, either an expert or an apprentice is available with equal probability. The expert's stationary policy T1 and the apprentice's stationary policy T3 are given. The goal of each model is to calculate a shared autonomy policy for the robot.

When an apprentice is available the robot can not do better than T3 so CoCAS selects T3. When an expert is available both T1 and T2 are optimal. But T2 requires fewer human assistants so CoCAS will select T2. On the other hand, SO-SAS will first calculate the robot's policy in isolation. The optimal isolated policy for the robot is T3. After that, it will try to combine the robot's and the expert's policies. For SO-SAS, the only way to get to the optimal destination when an expert is available is to transfer control at cell C1. while this also achieves optimal reward like CoCAS it creates unnecessary reliance on humans. Finally, the CAS model can not distinguish between expert and apprentice. Therefore, the CAS model will think the operator is successful $\sim 50\%$ of the time at navigating on icy cells. As a consequence, a CAS agent will always take T3. This is suboptimal when an expert is available.

4.3 Learning and Optimization for CoCAS

One of the crucial differences between the SO-SAS and CoCAS models is that the latter improves different components of its model by learning from experience over time. There are several learnable components in the CoCAS model including κ in the Autonomy model, and $\lambda, T_{\mathcal{H}}, \tau_{\mathcal{H}}$ in the Operator model. These components can be initialized with prior information, and physical or legal constraints [2]. The model and policy are updated after collecting a sufficient amount of new experience. Like the CAS model, CoCAS also employs a gated-exploration strategy, i.e., the agent needs explicit human permission to explore new autonomy levels. This ensures safety while gathering information about previously unexplored parts of the model.

In order to optimize the multidimensional reward function $\overline{R} = [R, \mu, \rho]$, several optimization approaches can be taken. In this paper, we assume a linear scalarization with function f parameterized by a weight vector $\mathbf{w}$ such that $f(\mathbf{w}, \overline{R}) = \mathbf{w}[R, \mu, \rho]^T$ [27]. With some modifications, the problem could be extended to handle both lexicographic [38] and constrained [1] models.

4.4 Properties

For consistency, we re-use the original notation from Basich et al. [2], and briefly re-state the primary properties introduced in [2]. λ-stationarity is the property of a CoCAS, in which the feedback profile, λ, has converged to the point where there is no expected value from acquiring new feedback from the human. Level-optimality is the condition in which a CoCAS, operates at its competence for a given state and action under its optimal policy, although the definition does not require that the competence is explicitly known by the system.

Unlike the cost-minimizing, goal-oriented SSP model used in the original CAS model [2], we model the CoCAS using arbitrary-reward maximizing MDPs, and restate the central definition of *competence* in this setting:

Definition 2. Let $\lambda^{\mathcal{H}} : \overline{S} \times \overline{A} \to \Delta^{|\Sigma|}$ be the stationary distribution of feedback signals for the N operators, $\mathcal{H}$. The **competence** of CoCAS $\mathcal{S}$, denoted $\chi_{\mathcal{S}}$, is a mapping from $\overline{S} \times A$ to the *reward-maximizing* level of autonomy given perfect knowledge of $\lambda^{\mathcal{H}}$. Formally:

$$\chi_{\mathcal{S}}(\bar{s}, a) = \operatorname*{argmax}_{l \in \mathcal{L}} q^*(\bar{s}, (a, l); \lambda^{\mathcal{H}}) \tag{1}$$

where $q^*(\bar{s}, (a, l); \lambda^{\mathcal{H}})$ is the cumulative expected reward under the optimal policy π^* when taking action $\bar{a} = (a, l)$ in state $\bar{s}$ conditioned on the feedback distribution $\lambda^{\mathcal{H}}$.

Note that we do not assume that $\lambda_{\mathcal{H}}$ is known by the CoCAS or even by the human explicitly.

It is straightforward to observe that, *under the same conditions stated* in Proposition 5.6 and Theorem 5.7 in [2], a CoCAS will also converge to λ-stationarity and level-optimality. The key observation here is that the addition of multiple stochastic operators, each of whom may have their own feedback profiles and cost functions, is captured by $S_{\mathcal{H}}$, which is included in the state representation $\overline{S}$, and hence under the assumption that every $(\bar{s}, \bar{a}) \in \overline{S} \times \overline{A}$ is visited sufficiently in the limit, the system will still learn the optimal level of autonomy for each operator in each operator state, which is always fully observed.

Proposition 1. Let $\mathcal{S}$ be a CoCAS and let $\lambda_t^{\bar{s},a}$ be the random variable representing $\lambda(\bar{s}, a)$ after having received t feedback signals for $(\bar{s}, a)$ where each signal is sampled from the true distribution $\lambda^{\mathcal{H}}(\bar{s}, a)$. As $t \to \infty$, if no $(\bar{s}, a)$ is starved, $\mathcal{S}$ will converge to λ-stationarity (see Def. 5.4 Basich et al. [2]).

Proof Sketch. Let the expected value of sample information (EVSI) on $\sigma \in \Sigma$ for $(\bar{s}, a)$ to be defined as in [2]. Fix $(\bar{s}, a)$. As each feedback signal for $(\bar{s}, a)$ is drawn from the true distribution $\lambda_{\mathcal{H}}(\bar{s}, a)$ i.i.d, then by a straightforward application of the law of large numbers, it follows that the sequence $\{\lambda_t^{(\bar{s},a)}\}$ will converge in distribution to $\lambda_{\mathcal{H}}^{(\bar{s},a)} = \mathbb{E}[\lambda_{\mathcal{H}}(\bar{s}, a)]$. Hence, it follows that $\lim_{t\to\infty} Pr[|\lambda_t^{\bar{s},a} - \lambda_{\mathcal{H}}^{\bar{s},a}]| > \epsilon] = 0 \; \forall \epsilon > 0$. Consequently, as $t \to \infty$, the probability that $\lambda_t^{\bar{s},a} = \lambda_{\mathcal{H}}^{\bar{s},a}$ defines a Dirac delta function with point mass centered at $\lambda_{\mathcal{H}}^{\bar{s},a}$. The rest of the proof follows the proof of Prop. 5.6 in [2]. □

Proposition 2. Let $\mathcal{S}$ be a CoCAS that follows any level-exploration strategy that ensures a non-zero probability that all reachable levels of autonomy are explored and switches to exploitation once λ-stationarity has been reached, and where $\chi_{\mathcal{S}}(\bar{s}, a)$ is reachable from κ_0 for all $(\bar{s}, a) \in \overline{S} \times A$. Then if no $(\bar{s}, a)$ is starved, as $t \to \infty$, $\mathcal{S}$ will converge to level-optimality.

Proof Sketch. The proof follows that of the proof of Thm. 5.7 in Basich et al. [2]. By Prop. 1, $\mathcal{S}$ will reach λ-stationarity, which ensures that for any $\bar{s} \in \overline{S}$ and action $a \in A$ the optimal level of autonomy $l^* \in \mathcal{L}$ is known in the limit, and hence we can determine in the limit the competence $\chi(\bar{s}, a)$ for all $\bar{s} \in \overline{S}$ and $a \in A$ so long as that level is explorable by the system (that is, both allowed under the human's true autonomy profile and reachable from the initial autonomy profile) with non-zero probability, which is true by assumption. □

A notable departure from the original theory of competence modeling in [2] is the additional dependence that *competence* has on the human operator in the CoCAS model. Whereas we previously assumed a completely stationary "human authority" that was perfectly consistent and non-variant in its performance and, critically, that the human could always bail out the system when necessary (if at a potentially high cost), that is not the case here. Rather, the human operator(s) themselves may vary in both their performance and reliability depending on their state or, more generally, the state of the human-agent connection. For example the operator's state may reflect performance degradation via becoming more tired after working for a long time, or alternatively variance in human-agent communication stability, e.g. how stable the connection is.

While it is obvious that CoCAS is a generalization of CAS as every CAS is a CoCAS with a single operator and a single state, it is also the case that all fully-observable SO-SAS (that is, all SO-SAS where the true operator states are fully observed at each state) can be modeled as a CoCAS where the action taken by the CoCAS is simply the selection operator in the context where each level of autonomy is fully controlled by a single operator (including the autonomous agent).

Proposition 3. Any fully-observable SO-SAS can be modeled as a CoCAS.

Proof. Let $M = \langle S^E, X, s_0, b_0, A, \Omega, T, O, R, \gamma \rangle$ be a fully- observable SO-SAS, i.e. so $\Omega = S^E \times X$ where S^E is the environment state space and $X = X_1 \times \cdots \times X_N$ is the operator profile state space, and the rest is as defined in [12]. Now, let $\mathcal{L} = \{l_1, ..., l_N\}$ be the levels of autonomy where l_i represents full control by the i^{th} operator, where the first operator is the autonomous agent, and let the CoCAS domain action set be $\emptyset$ such that $\overline{A} = \mathcal{L} \cong A$ i.e. the selection of which operator to be in control. Let $S = S^E$ and $S_{\mathcal{H}} = X$, so that $\overline{S} = S_{\mathcal{H}} \times S \times \mathcal{L} = X \times S^E \times \mathcal{L}$, and assume by construction that $\mathcal{L}$ does not impact either $\overline{T}$ or $\overline{R}$, so that we can simplify that $\overline{S} = S_{\mathcal{H}} \times S$. Then we have that $\overline{T} : \overline{S} \times \overline{A} \to \Delta^{\overline{S}} \cong T : (S^E \times X) \times A \to \Delta^{(S^E \times X)} \cong T : (S^E \times X) \times A \times (S^E \times X) \to [0, 1]$, and $\overline{R} : \overline{S} \times \overline{A} \to \mathbb{R} \cong R : (S^E \times X) \times A \to \mathbb{R}$. □

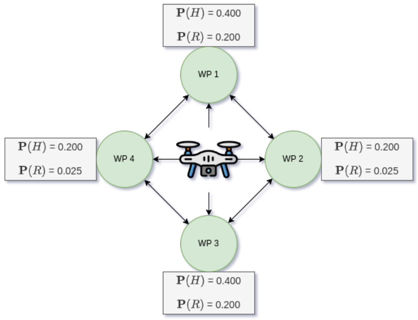


Figure 3: Illustration of the abstracted map for the unmanned aerial vehicle domain. $\mathrm{P}(H)$ and $\mathrm{P}(R)$ are the probabilities of taking a picture successfully by the human and the UAV.

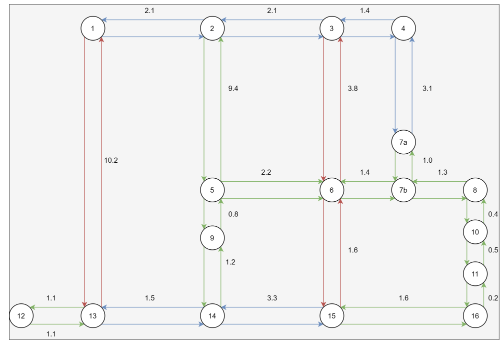


Figure 4: Illustration of the abstracted map for the autonomous vehicle delivery service domain.

5 EMPIRICAL EVALUATIONS

In this section we discuss the empirical evaluations performed to validate the proposed CoCAS model's efficacy against several other baseline models. We begin by describing the two simulated domains used to evaluate the CoCAS model — a UAV surveillance domain and an autonomous vehicle delivery domain. We then present the results of simulations performed in these domains using the CoCAS model and the following comparative benchmark models: *Operator* allows for no autonomy throughout execution and is entirely performed by a human operator (which may change during execution); *R-SO-SAS* randomly selects the operator uniformly at each timestep to act, where each operator follows a fixed pre-computed policy; *SO-SAS* follows the model from [12], but where the operator state is fully-observed at each timestep; *C-SO-SAS* follows the SO-SAS model and also includes the cost model to operators during planning; and *CAS* follows the model from [2].

5.1 Domain 1: UAV Surveillance

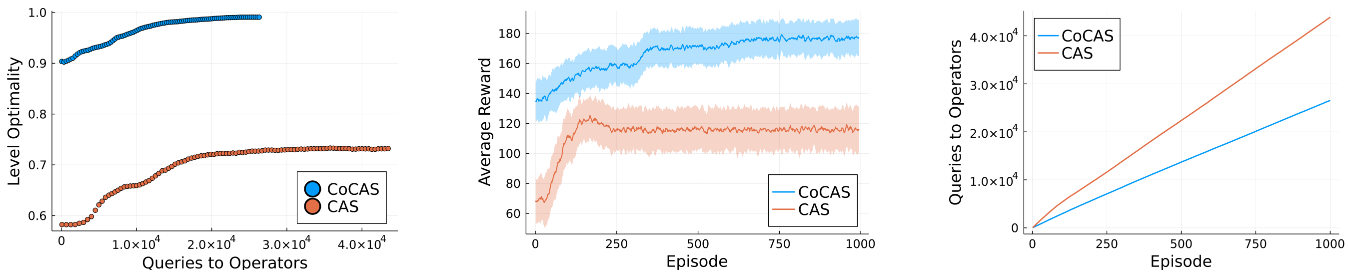
We apply our model to an extension of the UAV surveillance domain [12, 13] in which a UAV flies around a map to take photos of 4 pre-designated waypoints until all photos are of sufficient quality. Each waypoint has a different success rate of taking a good quality picture based on its difficulty. After taking pictures the UAV also have to verify from the human operator whether the pictures are sufficiently good or if more picture is needed. A state in the domain is represented by $\langle ID, \theta_1, \theta_2, \theta_3, \theta_4 \rangle$, where the *ID* indicates the current location (waypoint) of the UAV and $\theta_i \in \{NoPicture, UnverifiedPicture, VerifiedPicture\}$ represent of the current picture status of their corresponding waypoint. The agent has 7 actions available, $\{W_1, W_2, W_3, W_4,$ *TakePicture*, *Verify*, *EndMission*$\}$, where W_i indicates the action of going to waypoint i. While taking good pictures provides a positive reward, moving from one waypoint to another and taking pictures has an associated operation cost representing factors such as fuel and time.

We extend the domain in the following ways. First, we introduce three of the levels of autonomy used in [2]: *no autonomy* in which the UAV queries the active human operator to take the picture, *verified autonomy* in which the UAV queries the active human operator for verifying the quality photos and permission to end the mission, and *unsupervised autonomy* in which the UAV itself takes the photo itself and move around waypoints. Second, we include an extended operator model where an operator is assisting several UAVs at a time, and based on the demand, the human operator might have different availability. The operator state is represented by $S_{\mathcal{H}} = \{\theta_{available}, \theta_{PartiallyAvailable}, \theta_{Busy}\}$. Operator's availability affects both their ability to help the UAV and the cost of helping. In an available state, humans can help with all types of actions. In a partially available state, the operator can help with verification and mission end permission. Asking for taking pictures will require more cost representing the opportunity cost of not helping other UAVs. In a busy state, the operator is busy helping another UAV and can not help with any action. In general, in all waypoints, the human operator has a better or equal success rate of taking a quality picture than the autonomous UAV.

5.2 Domain 2: AV Delivery Service

In this domain, an autonomous vehicle (AV) is tasked with picking up and dropping off goods around an area, where its objective is to reach its destination in the most cost-effective manner. There are two types of state: node states representing intersections, and edge states representing road segments. Node states are represented by the tuple $\langle ID, p, o, v, \theta \rangle$, and edge states by the tuple $\langle u, v, \theta, o, l, r \rangle$. Here, *ID* is the ID of the node, p is 1 if there are pedestrians or else 0, v is the number of relevant other vehicles, θ is the AV's heading, u and v are the start and end node IDs, o is 1 if there is an obstruction in the AV's lane or else 0, l is the number of lanes on the road, and r denotes road restrictions if any.

The AV drives autonomously (l_0) but may seek assistance from human tele-operators: requesting approval for certain actions (l_1) or requesting full tele-operative control of the vehicle (l_0) (which may be denied). We model the existence of two human operators: a local operator, $\mathcal{H}_l$, and a global operator, $\mathcal{H}_g$ The local operator's connection is always stable allowing them to provide support for the AV in any capacity requested; however the local operator may become busy assisting another vehicle in the fleet when the demand is high. A global operator, however, is always available, but their connection may be either stable or unstable. We additionally consider contextual information, $\theta_C = \langle d, t, w \rangle \in \Theta_C$ where d is the current AV demand, t is the time of day, and w is



(a) (b) (c)

Figure 5: CAS and CoCAS performance over time in UAV Surveillance.

the current weather condition. Hence we get that $S_{\mathcal{H}} = \{\theta_{busy}, \theta_{active}\} \times \{\theta_{stable}, \theta_{unstable}\} \times \{\mathcal{H}_l, \mathcal{H}_g\} \times \Theta_C$. When stable, the global operator can provide verification for actions with consistency 0.8, and full tele-operation in certain non-challenging conditions; when unstable, the global operator can only provide verification for actions with consistency 0.7. We also require that the AV transfer control to a tele-operator when driving on roads that are demarcated as *pedestrian zones* or *school zones*.

We define $T_{\mathcal{H}}$ as follows: $\mathcal{H}_l$ can become busy or available with probabilities $1 - 0.5^d$ and 0.5^d at each step respectively. Note that, when there is zero demand, $\mathcal{H}_l$ is always available. At each step, $\mathcal{H}_g$'s connection quality changes with probability 0.25 and otherwise remains the same. We model the human cost function, ρ, to be based off of the expected opportunity cost incurred when the operator helps the AV and hence is unavailable to help another vehicle in the fleet. We apply this cost only to the local operator whose capacity is limited, and scale by the current fleet demand, d, but apply no such cost to the global operator, as we assume there is a sufficient supply to cover all demand at the lowered level of assistance. However, there is always a small baseline cost of requesting a transfer of control.

Models	Domain Reward	Human Cost	Total Reward
Operator	262.52(±0.58)	240.81(±1.08)	21.70(±0.95)
R-SO-SAS	210.95(±0.94)	162.2(±0.81)	48.82(±0.91)
SO-SAS	**285.81**(±0.50)	178.4(±0.95)	107.40(±0.80)
C-SO-SAS	227.80(±0.96)	52.35(±0.31)	174.90(±0.89)
CAS	260.01(±0.60)	143.7(±0.94)	116.00(±0.30)
CoCAS	212.58(±1.01)	**24.35(±0.10)**	**188.23(±0.95)**

Table 1: Results Summary from UAV Surveillance.

5.3 Empirical Results

Table 1 shows the results of our model (CoCAS) against the 5 baseline models described above on the UAV surveillance domain. We note that the results in this table are based on models that are fully aware of the system's true competence during planning to ensure a fair comparison between models. These results show that the CoCAS achieves the highest cumulative reward, as well as the lowest human cost over all models where autonomy is shared. We emphasize that while *C-SO-SAS* performs very well, it is still outperformed by *CoCAS* due to the phenomenon illustrated in Example 2. That is, optimally switching between policies computed optimally in isolation is not guaranteed to result in the optimal global policy. Finally, we see SO-SAS achieving the highest domain reward at the expense of the high human cost, resulting in a low total reward. This is because SO-SAS does not consider human cost at all, resulting in over-reliance on human operators.

Figure 5a plots the comparison in level-optimality as a function of episodes between the *CAS* and *CoCAS* levels as they learn their competence over time. We can see that the CoCAS model reaches 100% level optimality, whereas the standard CAS model only reaches 78%, as it is not able to properly converge feedback profile given that the human operator states have been effectively marginalized out, leading to an inability to learn its true competence in a large portion of the state space. Figure 5b plots the comparison in reward accrued by the CAS and CoCAS models as they learn their competence over time. Unsurprisingly, the CoCAS's reward grows with the increased competence, whereas the CAS's reward flattens out by episode 500 after which point it fails to improve its average performance. Table 2 shows the results of the CoCAS against the same 5 baseline models on the AV delivery service domain; unlike the UAV domain, the objective here is to minimize the cumulative cost. In particular, we can see that the CoCAS outperforms all other approaches in average cost across all 8 routes, and the lowest standard deviation in all but one case as well (Route 8).

Figures 6 depicts results in the learning simulation for both the CAS and CoCAS over 200 episodes. In the top set of figures, the contextual information (time of day and weather conditions) is fixed throughout all 200 episodes, whereas in the bottom set the contextual information varies across episodes. Figures 6a and 6d depicts the level optimality of *actions taken by the system* as a function of the number of queries made to the human. When the contextual information is fixed, we see that the CAS is able to converge to 100% level-optimality, although at a much slower rate than the COCAS, which reaches it very efficiently. When it is not fixed, the CoCAS can still reach 100% level-optimality over actions taken but the level-optimality of the CAS remains highly variable, and as low as 70% due to its inability to appropriately learn across unmodeled contextual information that alters the dynamics of the operators' feedback and interactions.

Figures 6b and 6e depict the mean and standard deviation of the incurred cost for both the CoCAS and CAS over the 200 episodes.

Models	Route 1	Route 2	Route 3	Route 4	Route 5	Route 6	Route 7	Route 8
Operator	67.09(±11.15)	207.31(±23.59)	118.86(±17.69)	169.39(±19.96)	110.75(±17.83)	182.83(±22.87)	160.37(±21.78)	96.91(±16.67)
R-SO-SAS	52.20(±43.73)	223.07(±54.83)	130.96(±45.69)	166.83(±42.61)	115.43(±42.86)	202.17(±59.32)	191.84(±62.82)	193.44(±44.71)
SO-SAS	42.08(±6.20)	168.76(±21.42)	87.71(±9.51)	127.44(±10.09)	76.06(±9.32)	136.38(±12.29)	111.31(±12.42)	72.96(±10.91)
C-SO-SAS	44.51(±4.59)	155.65(±13.72)	84.61(±8.52)	118.18(±7.67)	75.63(±6.35)	127.65(±9.87)	105.61(±11.80)	68.29(±9.75)
CAS	41.40(±6.62)	176.00(±25.62)	84.70(±77.90)	129.00(±15.30)	69.40(±4.79)	132.00(±11.98)	121.40(±24.77)	65.80(±**6.05**)
CoCAS	**39.71(±4.00)**	**142.85(±10.81)**	**77.93**(±7.56)	**112.87**(±7.39)	**68.44**(±3.89)	**119.98**(±9.71)	**99.80**(±9.55)	**61.35**(±6.85)

Table 2: Results Summary from AV Delivery Service.

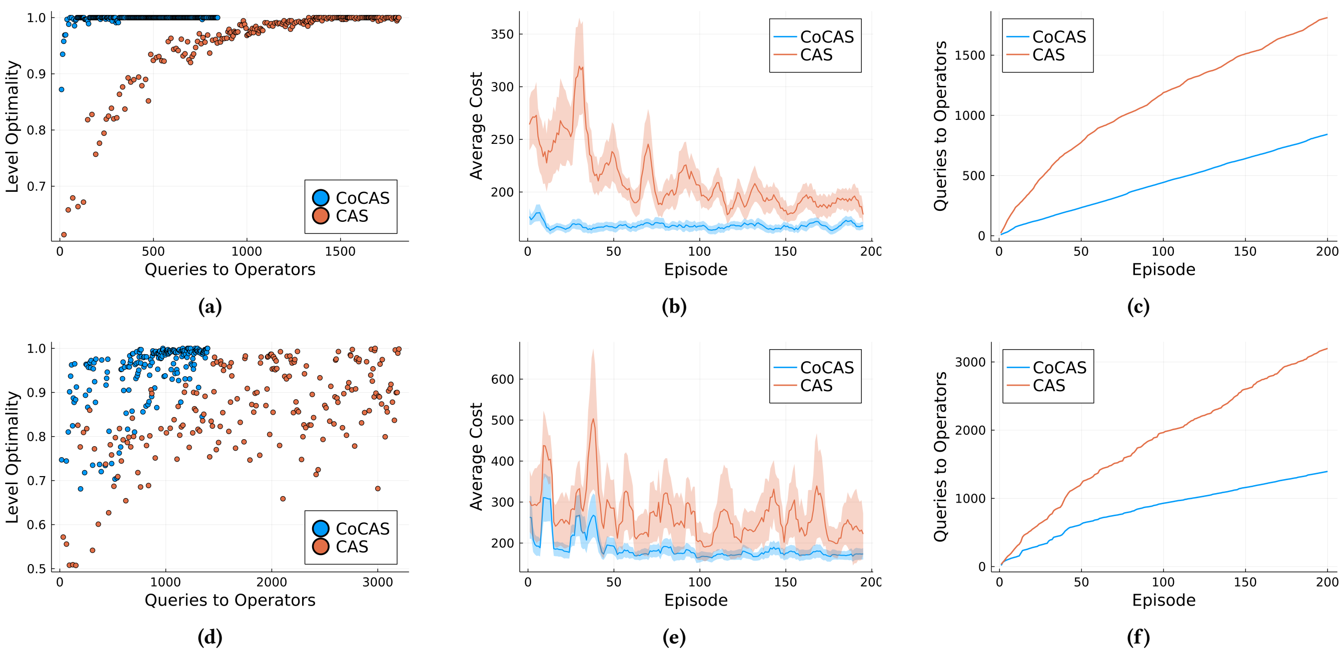


Figure 6: CAS and CoCAS performance in AV Delivery Service with static (top) and dynamic (bottom) contextual parameters.

Notably, the CoCAS quickly minimizes the incurred cost in both scenarios, and even in the fixed-context case where the CAS reaches 100% level-optimality in actions taken, the CoCAS is still more cost-effective in its operation. When the contextual information varies, where the CAS is unable to sufficiently learn its competence, its performance is both less cost-effective than the CoCAS and significantly more noisy. From figures 6c and 6f we see that the CAS queries the operators at roughly twice the rate as the CoCAS.

6 CONCLUSION

In this paper, we introduce the CoCAS model for decision-theoretic planning in the context of multiple levels of autonomy and multiple stochastic heterogeneous human operators. Although we proved that a CoCAS can capture the SO-SAS model, we emphasize that each model has a distinct purpose. The SO-SAS model treats each operator as a separate-but-equal agent in the system and assumes that there is a known fixed policy for each operator based on complete knowledge of the domain, leaving the only decision to be which operator should be active at any given time. In the CoCAS model, there is a single executor that can utilize human operators as a *resource* to achieve their objective. The CoCAS model is particularly relevant in the case where decisions about the level of autonomy (and consequently transfer-of-control) are made in tandem with the executable action decisions themselves, and in domains where the agent's true competence may not be known *a priori* such as the CAS model [2].

We present a generalization of the *human feedback model* defined in our prior work [2], by integrating contextual information and stochastic operators into the competence-modeling framework. Although it is outside the scope of this paper, a similar generalization could be made to the *autonomy model*, most obviously by extending constraints on autonomy to depend on contextual information and the state of the operators. Additionally, natural directions for future research include extending the CoCAS to partially-observable settings both in the domain model as well as the operator model, learning the operator model parameters online, and formulating the fleet-wide human-to-agent matching problem to optimize a fleet of agents that rely on occasional human assistance.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation grant IIS-1954782 and the Alliance Innovation Lab Silicon Valley.

REFERENCES

[1] Eitan Altman. 1999. *Constrained Markov Decision Processes: Stochastic Modeling.* Routledge.

[2] Connor Basich, Justin Svegliato, Kyle Hollins Wray, Stefan Witwicki, Joydeep Biswas, and Shlomo Zilberstein. 2020. Learning to optimize autonomy in competence-aware systems. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS).* 123–131.

[3] Jacob Beal and Miles Rogers. 2020. Levels of autonomy in synthetic biology engineering. *Molecular Systems Biology* 16, 12 (2020), e10019.

[4] Jenay M. Beer, Arthur D. Fisk, and Wendy A. Rogers. 2014. Toward a framework for levels of robot autonomy in human-robot interaction. *Journal of Human-Robot Interaction* 3, 2 (2014), 74.

[5] Dimitri P. Bertsekas and John N. Tsitsiklis. 1991. An analysis of stochastic shortest path problems. *Mathematics of Operations Research* 16, 3 (1991), 580–595.

[6] Joydeep Biswas and Manuela Veloso. 2016. The 1,000-km challenge: Insights and quantitative and qualitative results. *IEEE Intelligent Systems* 31, 3 (2016), 86–96.

[7] Alberto Broggi, Massimo Bertozzi, Alessandra Fascioli, C. Guarino Lo Bianco, and Aurelio Piazzi. 1999. The ARGO autonomous vehicle's vision and control systems. *International Journal of Intelligent Control and Systems* 3, 4 (1999), 409–441.

[8] Alberto Broggi, Pietro Cerri, Mirko Felisa, Maria Chiara Laghi, Luca Mazzei, and Pier Paolo Porta. 2012. The VisLab intercontinental autonomous challenge: An extensive test for a platoon of intelligent vehicles. *International Journal of Vehicle Autonomous Systems* 10, 3 (2012), 147–164.

[9] David J. Bruemmer, Douglas A. Few, Ronald L. Boring, Julie L. Marble, Miles C. Walton, and Curtis W. Nielsen. 2005. Shared understanding for collaborative control. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 35, 4 (2005), 494–504.

[10] Manolis Chiou, Nick Hawes, and Rustam Stolkin. 2021. Mixed-Initiative variable autonomy for remotely operated mobile robots. *ACM Transactions on Human-Robot Interaction (THRI)* 10, 4 (2021), 1–34.

[11] Manolis Chiou, Nick Hawes, Rustam Stolkin, Kimron L Shapiro, Jess R Kerlin, and Andrew Clouter. 2015. Towards the principled study of variable autonomy in mobile robots. In *IEEE International Conference on Systems, Man, and Cybernetics (ICSMC).* IEEE, 1053–1059.

[12] Clarissa Costen, Marc Rigter, Bruno Lacerda, and Nick Hawes. 2022. Shared autonomy systems with stochastic operator models. In *International Joint Conference on Artificial Intelligence (IJCAI).* 4614–4620.

[13] Lu Feng, Clemens Wiltsche, Laura R. Humphrey, and Ufuk Topcu. 2016. Synthesis of human-in-the-loop control protocols for autonomous systems. *IEEE Transactions on Automation Science and Engineering* 13 (2016), 450–462.

[14] George Ferguson, James F. Allen, and Bradford W. Miller. 1996. TRAINS-95: Towards a Mixed-Initiative Planning Assistant. In *Third Conference on Artificial Intelligence Planning Systems (AIPS).* 70–77.

[15] Fanny Ficuciello, Guglielmo Tamburrini, Alberto Arezzo, Luigi Villani, and Bruno Siciliano. 2019. Autonomy in surgical robots and its meaningful human control. *Journal of Behavioral Robotics* 10, 1 (2019), 30–43.

[16] Terrence Fong, Charles Thorpe, and Charles Baur. 2003. Multi-robot remote driving with collaborative control. *IEEE Transactions on Industrial Electronics* 50, 4 (2003), 699–704.

[17] Yang Gao and Steve Chien. 2017. Review on space robotics: Toward top-level science through space exploration. *Science Robotics* 2, 7 (2017).

[18] E. Amir M. Ghalamzan, Firas Abi-Farraj, Paolo Robuffo Giordano, and Rustam Stolkin. 2017. Human-in-the-loop optimisation: mixed initiative grasping for optimally facilitating post-grasp manipulative actions. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* IEEE, 3386–3393.

[19] Nathan A. Greenblatt. 2016. Self-driving cars and the law. *IEEE Spectrum* 53, 2 (2016), 46–51.

[20] Eric Krotkov, Reid Simmons, Fabio Cozman, and Sven Koenig. 1996. Safeguarded teleoperation for lunar rovers: From human factors to field trials. In *IEEE Planetary Rover Technology and Systems Workshop*, Vol. 26. 28.

[21] Patrick Lin. 2016. Why ethics matters for autonomous cars. *Autonomous Driving: Technical, Legal and Social Aspects* (2016), 69–85.

[22] Markus Maurer, J. Christian Gerdes, Barbara Lenz, and Hermann Winner. 2016. *Autonomous driving: Technical, legal and social aspects.* Springer.

[23] John F. Mustard, David W. Beaty, and Deborah S. Bass. 2013. Mars 2020 science rover: Science goals and mission concept. In *AAS Division for Planetary Sciences Annual Meeting*, Vol. 45. 211–217.

[24] Raja Parasuraman, Thomas B. Sheridan, and Christopher D. Wickens. 2000. A model for types and levels of human interaction with automation. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 30, 3 (2000), 286–297.

[25] Giannis Petousakis, Manolis Chiou, Grigoris Nikolaou, and Rustam Stolkin. 2020. Human operator cognitive availability aware Mixed-Initiative control. In *IEEE International Conference on Human-Machine Systems (ICHMS).* IEEE, 1–4.

[26] Martin L. Puterman. 1990. Markov decision processes. *Handbooks in Operations Research and Management Science* 2 (1990), 331–434.

[27] Diederik M. Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. 2013. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research* 48 (2013), 67–113.

[28] Stephanie Rosenthal, Joydeep Biswas, and Manuela M. Veloso. 2010. An effective personal mobile robot agent through symbiotic human-robot interaction. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, Vol. 10. 915–922.

[29] SAE On-Road Automated Vehicle Standards Committee. 2014. Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems. *SAE Standards Journal* 3016 (2014), 1–16.

[30] Wilko Schwarting, Javier Alonso-Mora, and Daniela Rus. 2018. Planning and decision-making for autonomous vehicles. *Annual Review of Control, Robotics, and Autonomous Systems* 1, 1 (2018), 187–210.

[31] Thomas B. Sheridan. 1992. *Telerobotics, Automation, and Human supervisory control.* Cambridge, MA: MIT Press.

[32] Manuela Veloso, Joydeep Biswas, Brian Coltin, Stephanie Rosenthal, Tom Kollar, Cetin Mericli, Mehdi Samadi, Susana Brandao, and Rodrigo Ventura. 2012. Cobots: Collaborative robots servicing multi-floor buildings. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* 5446–5447.

[33] Manuela M. Veloso, Joydeep Biswas, Brian Coltin, and Stephanie Rosenthal. 2015. CoBots: Robust symbiotic autonomous mobile service robots. In *International Joint Conference on Artificial Intelligence (IJCAI).* 4423–4429.

[34] Michael T. Wolf, Christopher Assad, Matthew T. Vernacchia, Joshua Fromm, and Henna L. Jethani. 2013. Gesture-based robot control with variable autonomy from the JPL BioSleeve. In *IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 1160–1165.

[35] Kyle Hollins Wray, Luis Enrique Pineda, and Shlomo Zilberstein. 2016. Hierarchical approach to transfer of control in semi-autonomous systems. In *International Joint Conference on Artificial Intelligence (IJCAI).* 517–523.

[36] Kyle Hollins Wray, Stefan J. Witwicki, and Shlomo Zilberstein. 2017. Online decision-making for scalable autonomous systems. In *International joint conference on artificial intelligence.* 4768–4774.

[37] Kyle Hollins Wray and Shlomo Zilberstein. 2015. Multi-Objective POMDPs with Lexicographic Reward Preferences. In *International Joint Conference on Artificial Intelligence (IJCAI).* 1719–1725.

[38] Kyle Hollins Wray, Shlomo Zilberstein, and Abdel-Illah Mouaddib. 2015. multi-objective MDPs with conditional lexicographic reward preferences. In *AAAI Conference on Artificial Intelligence (AAAI).* 3418–3424.

[39] Guang-Zhong Yang, James Cambias, Kevin Cleary, Eric Daimler, James Drake, Pierre E. Dupont, Nobuhiko Hata, Peter Kazanzides, Sylvain Martel, Rajni V. Patel, et al. 2017. Medical robotics—Regulatory, ethical, and legal considerations for increasing levels of autonomy. *Science Robotics* 2, 4 (2017), 8638.

[40] Michael Yip and Nikhil Das. 2019. Robot autonomy for surgery. In *The Encyclopedia of MEDICAL ROBOTICS: Volume 1 Minimally Invasive Surgical Robotics.* World Scientific, 281–313.

[41] Shlomo Zilberstein. 2015. Building strong semi-autonomous systems. In *AAAI Conference on Artificial Intelligence (AAAI).* 4088–4092.