Synthesizing least-limiting guidelines for safety of semi-autonomous systems

Jana Tumova and Dimos V. Dimarogonas

Abstract— **We consider the problem of synthesizing safe-by-design control strategies for semi-autonomous systems. Our aim is to address situations when safety cannot be guaranteed solely by the autonomous, controllable part of the system and a certain level of collaboration is needed from the uncontrollable part, such as the human operator. In this paper, we propose a systematic solution to generating least-limiting guidelines, i.e. the guidelines that restrict the human operator as little as possible in the worst-case long-term system executions. The algorithm leverages ideas from 2-player turn-based games.**

I. Introduction

Recent technological developments have enhanced the application areas of autonomous and semi-autonomous cyber-physical systems to a variety of everyday scenarios from industrial automation to transportation and to housekeeping services. These examples have a common factor; they involve operation in an uncertain environment in the presence of highly unpredictable and uncontrollable agents, such as humans. In robot-aided manufacturing, there is a natural combination of autonomy and human contribution. In semi-autonomous driving, the vehicle is partially controlled automatically and partially by a human driver. Even in fully autonomous driving, passengers and pedestrians interact with the vehicle and actively influence the overall system safety and performance. The need for obtaining guarantees on behaviors of these systems is then even more crucial as the stakes are high. Formal verification and formal methods-based synthesis techniques were designed to provide such guarantees and recently, they have gained a considerable amount of popularity in applications to correct-by-design robot control. For instance, in [12], [18] temporal logic control of robots in uncertain, reactive environments was addressed. In [11] control synthesis for nondeterministic systems from temporal logic specifications was developed. Loosely speaking, these works achieve the provable guarantees by accounting for the *worst-case* scenarios in the control synthesis procedure. The uncertainty is therein treated as an *adversary*, which however, often prevents the synthesis procedure to find a correct-by-design autonomous controller.

In this paper, we take a fresh perspective on correct-by-design control synthesis. We specifically focus on situations when the desired controller does not exist. In contrast to the above mentioned approach, we view the uncertain, uncontrollable elements in the system as *collaborative* in the sense that they have as much interest in keeping the overall system behavior safe, effective, and efficient as the autonomous controller does. At the same time, we still view them as to a large extent *uncontrollable* in the sense that they still have their own intentions and we cannot force them to follow literal step-by-step instructions. In contrast, we aim to *advise* them on what not to do if completely necessary, while keeping their options as rich as possible.

For example, consider a collaborative human-robot manufacturing task with the goal of assembling products ABC through connecting pieces of types A and C to a piece of type B. The human operator can put together A with B or with BC, whereas the autonomous robot can put together B or AB with C. Our goal is to guarantee system safety, meaning that the human and the robot do not work with the same piece of type B at the same time. While we can design a controller for the robot that does not reach for a piece being held by a human, we cannot guarantee that the human will not reach for a piece being held by the robot. To that end, we aim to synthesize *guidelines* for the human, i.e. advise that reaching for a piece that the robot holds will lead to the safety violation. Under the assumption that the human follows this advise, the safety is guaranteed. Yet, this advise is still much less restrictive for the human operator than if the human-robot system was considered controllable as a whole. Namely, in such a case, a correct-by-design controller could dictate the human to always touch only solo B pieces while the robot would be supposed to work only with AB pieces pre-produced by the human. Clearly, the former mentioned guidelines allow for much more freedom of the human's decisions as the human may choose to work with an instance of B piece or BC piece. A similar situation occurs in an autonomous driving scenario with a pedestrian crossing the street. If the pedestrian jumps right in front of the car, the collision is unavoidable. A possible guideline for the human enabling the system safety would be not to ever cross the street. This is however a very limiting constraint. Instead, advising the human not to cross the street if the car is close seems quite reasonable.

This paper introduces a *systematic way to synthesize least-limiting guidelines* for the uncontrollable elements in (semi-)autonomous systems, such as humans in human-robot systems, that allow the autonomous part of the system to maintain safety. Similarly as in some related work on correct-by-design control synthesis (e.g., [11]), we model the overall system state space as a two-player game on a graph with a safety winning condition. The autonomous, controllable entity takes the role of the game protagonist, whereas the un-

The authors are with the ACCESS Linnaeus Center, School of Electrical Engineering, KTH Royal Institute of Technology, SE-100 44, Stockholm, Sweden and with the KTH Centre for Autonomous Systems. This work was supported by the H2020 ERC Starting Grant BUCOPHSYS. {tumova,dimos}@kth.se

controllable entity is the adversary. We specifically work with situations, where the protagonist does not have a winning strategy in the game. We formalize the notion of *adviser* as a function that "forbids" the application of certain adversary's inputs in certain system states. Furthermore, we classify the advisers based on the level of limitation they impose on the adversary. Finally, we provide an algorithm to find a least-limiting adviser that allows the protagonist to win the game, i.e. to keep the system safe. We also discuss the use of the synthesized advisers for on-the-fly guidance of the system execution. In this work, we do not focus on how the interface between the adviser and the uncontrollable element, such as human, should look like. Rather than that, the contribution of this paper can be summarized as the development of a theoretical framework for automated synthesis of reactive, least-limiting guidelines and control strategies that guarantee the system safety.

Related work includes literature on synthesis of environment assumptions that enable a winning game [6] and on using counter-strategies for synthesizing assumptions in generalized reactivity (1) (GR(1)) fragment of LTL [13], [1]. These works however synthesize the assumptions in the form of logic formulas, whereas we focus on guiding the adversary through explicitly enumerating the inputs that should not be applied. Synthesis of maximally permissive strategies is considered in [4] and also in discrete-event systems literature in [17], where however, only controllable inputs are being restricted. Our approach is different to the above works, since we aim for systematic construction of reactive guidelines in the sense that if the least-limiting adviser is not followed, a suitable substitute adviser is supplied if such exists. We also use a different criterion to measure the level of limitation that is the worst-case long-term average of restrictions as opposed to the cumulative number of restrictions considered in [6] or the size of the set of behaviors considered in [4]. Other related literature studies problems of minimal model repair [3], [7], synthesis of least-violating strategies [9], [16], or design of reward structures for decision-making processes in context of human-machine interaction [14]. This work can be also viewed in the context of literature aimed at collaborative human-robot control, e.g., [15], [10].

The paper is structured as follows. In Sec. II we introduce necessary notation and preliminaries. In Sec. III, we state our problem. In Sec. IV, we introduce the synthesis algorithm in details and discuss the use of the synthesized solution for on-the-fly guidance. Sec. V concludes the paper and discusses future research. Throughout the paper, we provide several illustrative examples demonstrating the developed theory.

II. Preliminaries

Given a set S, we use 2^{S}, $|\mathsf{S}|$, S^*, S^ω to denote the powerset of S, the cardinality of S, and the set of all finite and infinite sequences of elements from S, respectively. Given a finite sequence w and a finite or an infinite sequence w', we use $w{\cdot}w'$ to denote their concatenation. Let $w(i)$ and $w_{\rightsquigarrow j}$ denote the i-th element of word w and the prefix of w that ends in $w(j)$, respectively. Furthermore, assuming that S is a set of finite sequences and S' is a set of finite and/or infinite sequences, $\mathsf{S} \cdot \mathsf{S}' = \{w \cdot w' \mid w \in \mathsf{S} \wedge w' \in \mathsf{S}'\}$. $\mathbb{Z}$ denotes the set of integers.

Definition 1 (Arena) *A 2-player turn-based game arena is a transition system $\mathcal{T} = (S, \langle S_p, S_a \rangle, s_{init}, U_p, U_a, T)$, where S is a nonempty, finite set of states; $\langle S_p, S_a \rangle$ is a partition of S into the set of protagonist (player p) states S_p and the set of adversary (player a) states S_a, such that $S_p \cap S_a = \emptyset$, $S_p \cup S_a = S$; $s_{init} \in S_p$ is the initial protagonist state; U_p is the set of inputs of the protagonist; U_a is the set of inputs of the adversary; $T = T_p \cup T_a$, is a partial injective transition function, where $T_p : S_p \times U_p \to S_a$ and $T_a : S_a \times U_a \to S_p$.*

Note that in a protagonist state, only an input of the protagonist can be applied, and analogously, in an adversary state, only an input of the adversary can be applied. We assume that from a protagonist state, the system can only transition to an adversary state and vice versa. This assumption is not restrictive, since it can be easily shown that any game arena with $T_p : S_p \times U_p \to S$ and $T_a : S_a \times U_a \to S$ can be transformed to satisfy it. Loosely speaking, each transition from a protagonist state to a protagonist state is split into two transitions, to and from a new adversary state. Analogous transformation can be applied to the transitions from adversary states to adversary states.

Let $U_i^{s_i} = \{u_i \in U_i \mid T_i(s_i, u_i) \text{ is defined}\}$ denote the set of inputs of player $i \in \{p, a\}$ that are *enabled* in the state $s_i \in S_i$. Arena $\mathcal{T}$ is *non-blocking* if $|U_i^{s_i}| \geq 1$, for all $i \in \{p, a\}$ and all $s_i \in S_i$ and *blocking* otherwise. A *play* in $\mathcal{T}$ is an *infinite* alternating sequence of protagonist and adversary states $\pi = s_{p,1}s_{a,1}s_{p,2}s_{a,2}\ldots$, such that $s_{p,1} = s_{init}$ and for all $j \geq 1$ there exist $u_{p,j} \in U_p, u_{a,j} \in U_a$, such that $T_p(s_{p,j}, u_{p,j}) = s_{a,j}$, and $T_a(s_{a,j}, u_{a,j}) = s_{p,j+1}$. Note that for each play π, $\pi(2k) \in S_a$, while $\pi(2k-1) \in S_p$, for all $1 \leq k$. A *play prefix* $\pi_{\rightsquigarrow j} = \pi(1)\ldots\pi(j)$ is a finite prefix of a play $\pi = \pi(1)\pi(2)\ldots$. Let $\mathit{Plays}^{\mathcal{T}}$ denote the set of all plays in $\mathcal{T}$. If a set of plays $\mathit{Plays}^{\dot{\mathcal{T}}}$ of a blocking arena $\dot{\mathcal{T}}$ is nonempty, then $\dot{\mathcal{T}}$ can be transformed into an equivalent non-blocking arena $\mathcal{T}$ via a systematic removal of *blocking states* and their adjacent transitions that are defined inductively as follows: (i) each $s_i \in S_i$, $i \in \{p, a\}$, such that $U_i^{s_i} = \emptyset$ is a blocking state and (ii) if $T_i(s_i, u_i)$ is a blocking state for each $u_i \in U_i^{s_i}$, then s_i, $i \in \{i, p\}$ is a blocking state, too. Then $\mathit{Plays}^{\dot{\mathcal{T}}} = \mathit{Plays}^{\mathcal{T}}$.

A *deterministic control strategy* (or strategy, for short) of player $i \in \{p, a\}$ is a partial function $\sigma_i^{\mathcal{T}} : S^* \cdot S_i \to U_i$ that assigns a player i's enabled input $u_i \in U_i^{s_i}$ to each play prefix in $\mathcal{T}$ that ends in a player i's state $s_i \in S_i$. Strategies $\sigma_p^{\mathcal{T}}, \sigma_a^{\mathcal{T}}$ *induce a play* $\pi^{\sigma_p^{\mathcal{T}}, \sigma_a^{\mathcal{T}}} = s_{p,1}s_{a,1}s_{p,2}s_{a,2}\ldots \in (S_p \cdot S_a)^\omega$, such that $s_{p,1} = s_{init}$, and for all $j \geq 1$, $T_p(s_{p,j}, \sigma_p(s_{p,1}s_{a,1}\ldots s_{p,j})) = s_{a,j}$, and $T_a(s_{a,j}, \sigma_a(s_{p,1}s_{a,1}\ldots s_{p,j}s_{a,j})) = s_{p,j+1}$. A strategy $\sigma_i^{\mathcal{T}}$ is called *memoryless* if it satisfies the property that $\sigma_i^{\mathcal{T}}(s_1 \ldots s_n) = \sigma_i^{\mathcal{T}}(s'_1 \ldots s'_m)$ whenever $s_n = s'_m$. Hence, with a slight abuse of notation, memoryless control strategies are viewed as functions $\varsigma_i^{\mathcal{T}} : S_i \to U_i$. The set of all strategies of player i in $\mathcal{T}$ is denoted by $\Sigma_i^{\mathcal{T}}$. The set of all plays induced by all strategies in $\Sigma_p^{\mathcal{T}}, \Sigma_a^{\mathcal{T}}$, i.e. the set of all plays in $\mathcal{T}$ is $Plays^{\Sigma_p^{\mathcal{T}}, \Sigma_a^{\mathcal{T}}} = \{\pi^{\sigma_p^{\mathcal{T}}, \sigma_a^{\mathcal{T}}} \mid \sigma_p^{\mathcal{T}} \in \Sigma_p^{\mathcal{T}}, \sigma_a^{\mathcal{T}} \in \Sigma_a^{\mathcal{T}}\}$. Analogously, we use $Plays^{\sigma_p^{\mathcal{T}}, \Sigma_a^{\mathcal{T}}} = \{\pi^{\sigma_p^{\mathcal{T}}, \sigma_a^{\mathcal{T}}} \mid \sigma_a^{\mathcal{T}} \in \Sigma_a^{\mathcal{T}}\}$ to denote the set of plays induced by a given strategy $\sigma_p^{\mathcal{T}}$ and by all strategies $\sigma_a^{\mathcal{T}} \in \Sigma_a^{\mathcal{T}}$.

A *game* $G = (\mathcal{T}, W)$ consists of a game arena $\mathcal{T}$ and a *winning condition* $W \subseteq Plays^{\Sigma_p^{\mathcal{T}}, \Sigma_a^{\mathcal{T}}}$ that is in general a subset of plays in $\mathcal{T}$. A *safety winning condition* is $W_{Safe} = \{\pi \in Plays^{\Sigma_p^{\mathcal{T}}, \Sigma_a^{\mathcal{T}}} \mid \text{ for all } j \geq 1.\, \pi(j) \in Safe\}$, where $S = \langle Safe, Unsafe \rangle$ is a partition of the set of states into the safe and unsafe state subsets. A protagonist's strategy $\sigma_p^{\mathcal{T}}$ is winning if $Plays^{\sigma_p^{\mathcal{T}}, \Sigma_a^{\mathcal{T}}} \subseteq W$. Let $\Omega_p^{\mathcal{T}} \subseteq \Sigma_p^{\mathcal{T}}$ denote the set of all protagonist's winning strategies.

Let $\mathcal{T} = (S, \langle S_p, S_a \rangle, s_{init}, U_p, U_a, T)$ be an arena and $w : S \times S \to \mathbb{Z}$ be a weight function that assigns a weight to each (s, s'), such that there exists $u \in U_p \cup U_a$, where $(s, u, s') \in T$. Then $(\mathcal{T}, w)$ can be viewed as an arena of a *mean-payoff game*. The value secured by protagonist's strategy $\sigma_p^{\mathcal{T}}$ is

$$\nu(\sigma_p^{\mathcal{T}}) = \inf_{\sigma_a^{\mathcal{T}} \in \Sigma_a^{\mathcal{T}}} \liminf_{n \to \infty} \frac{1}{n} \sum_{j=1}^{n} w(\pi^{\sigma_p^{\mathcal{T}}, \sigma_a^{\mathcal{T}}}(j), \pi^{\sigma_p^{\mathcal{T}}, \sigma_a^{\mathcal{T}}}(j+1)).$$

An *optimal protagonist's strategy* $\sigma_p^{\mathcal{T}*}$ secures the optimal value $\nu(\sigma_p^{\mathcal{T}*}) = \sup_{\sigma_p^{\mathcal{T}} \in \Sigma_p^{\mathcal{T}}} \nu(\sigma_p^{\mathcal{T}})$. Several algorithms exist to find the optimal protagonist's strategy, see, e.g., [5]. For more details on games on graphs in general, we refer the interested reader e.g., to [2].

III. Problem formulation

The *system* that we consider consists of two entities: the first one is the autonomous part of the system that we aim to control (e.g., a robotic arm), and the second one is the agent that is uncontrollable, and to a large extent unpredictable (e.g., a human operator in a human-robot manufacturing scenario). The overall state of such system is determined by the system states of these entities (e.g., the positions of the robotic and the human arms and objects in their common workspace and the status of the manufacturing). In this paper, we consider systems with a finite number of states Q (obtained, e.g., by partitioning the workspace into cells). The system state can change if one of the entities takes a decision and applies an input (e.g., the robot can move the arm from on cell to another, or the human can pick up an object). For simplicity, we assume that the entities take regular turns in applying their inputs. This assumption is however not too restrictive as we may allow the entities to apply a special pass input ϵ that does not induce any change to the current system state.

To model the system formally, we call the former, controllable entity the protagonist, the latter, uncontrollable entity the adversary, and we capture the impacts of their inputs to the system states through a game arena (see Def. 1)

$$\mathcal{T} = (S, \langle S_p, S_a \rangle, s_{init}, U_p, U_a, T). \qquad (1)$$

The set of the arena states is $S = Q \times \{p, a\}$ and each arena state $s = (q, i) \in S$ is defined by the system state $q \in Q$ and the entity $i \in \{p, a\}$ whose turn it is to apply its input, i.e. $(q, p) \in S_p$, and $(q, a) \in S_a$, for all $q \in Q$. Behaviors of the system are thus captured through plays in the arena.

The goal of the former, controllable entity is to keep the system safe, i.e. to avoid the subset of unsafe system states, while the latter entity has its own goals, such as to reach a certain system state, etc. Formally, the protagonist is given a partition of states $S = \langle Safe, Unsafe \rangle$ and the corresponding safety winning condition W_{Safe}. The arena $\mathcal{T}$ together with the safety winning condition W_{Safe} establish a game (T, W_{Safe}).

Example 1 *Consider the simplified manufacturing scenario outlined in the introduction. A system state is determined by the current pieces in the workspace and their status; each of them is either on the desk, held by the human, or by the robot:* $Q \subseteq 2^{\{A,B,C,AB,BC,ABC\} \times \{desk, human, robot\}}$. *The robot acts as the protagonist and the human as the adversary.* $s_{init} = \big(\{(A, desk), (B, desk), (C, desk)\}, a\big)$ *is an example of a system initial state. The inputs of the robot are* $U_p = \big\{\{grab_p, drop_p\} \times \{A, B, C, AB, BC, ABC\} \cup \{connect_p\} \times \{(B, C), (AB, C)\}\big\}$ *and similarly,* $U_a = \big\{grab_a, drop_a\} \times \{A, B, C, AB, BC, ABC\} \cup \{connect_a\} \times \{(A, B), (A, BC)\}\big\}$. *The transition function reflects the effect of inputs on the system state. For instance,*
$T\big(\big(\{(A, desk), (B, desk), (C, desk)\}, a\big), (grab_a, A)\big) =$
$= \big(\{(A, human), (B, desk), (C, desk)\}, p\big)$*, or*
$T\big(\big(\{(A, desk), (B, robot), (C, robot)\}, p\big), \big(connect_p, (B, C)\big)\big)$
$= \big(\{(A, human), (AB, robot)\}, a\big)\big)$.

Note that the transition function does not have to be manually enumerated. Rather than that, it can be generated from conditions, such as $T\big(\big(\{(x, y)\} \cup Z, a\big), (grab_a, x)\big) = \big(\{(x, human)\} \cup Z, p\big)$*, applied to all* $x \in \{A, B, C, AB, AC, ABC\}, y \in \{desk, robot\}, Z \subseteq (\{A, B, C, AB, AC, ABC\} \setminus \{x\}) \times \{desk, human, robot\}$.

The problem of finding a protagonist's winning control strategy $\sigma_p^{\mathcal{T}}$ guaranteeing system safety has been studied before and even more complex winning conditions have been considered [2]. In this work, we focus on a situation when the protagonist *does not* have a winning control strategy. For such cases, we aim to generate a least-limiting subset of adversary's control strategies that would permit the protagonist to win. Loosely speaking, this subset can be viewed as the minimal guidelines for the adversary's collaboration.

Note that this problem differs from the supervisory control of discrete event systems as we do not limit only the application of controllable, but also the uncontrollable inputs. However, it also differs from the synthesis of controllers for fully controllable systems as we aim to limit the adversary's application of uncontrollable inputs as little as possible. We formalize the guidelines for the adversary's collaboration through the notion of adviser and adviser restricted arena.

Definition 2 (Adviser) *An adviser is a mapping* $\alpha : S_a \to 2^{U_a}$*, where* $\alpha(s_a) \subseteq U_a^{s_a}$ *represents the subset of adversary's inputs that are forbidden in state* s_a.

Given an arena $\mathcal{T} = (S, \langle S_p, S_a\rangle, s_{init}, U_p, U_a, T_p \cup T_a)$, and an adviser α, the adviser restricted arena is $\dot{\mathcal{T}}^\alpha = (S, \langle S_p, S_a\rangle, s_{init}, U_p, U_a, \dot{T}_p^\alpha \cup \dot{T}_a^\alpha)$, where $\dot{T}_p^\alpha = T_p$ and $\dot{T}_a^\alpha = T_a \setminus \{(s_a, u_a, s_p) \mid u_a \in \alpha(s_a)\}$. The set of all plays in $\dot{\mathcal{T}}^\alpha$ is denoted by $Plays^{\dot{\alpha}}$.

If $\alpha(s_a) = U_a^{s_a}$ for some $s_a \in S_a$, the adviser restricted arena becomes blocking, and hence, not every sequence $s_{p,1}s_{a,1}s_{p,2}s_{a,2}\ldots s_{a,k}$, satisfying $s_{p,1} = s_{init}$, and for all $1 \le j \le k$, $1 \le \ell < k$, $\dot{T}_p^\alpha(s_{p,j}, \sigma_p(s_{p,1}s_{a,1}\ldots s_{p,j})) = s_{a,j}$, and $\dot{T}_a^\alpha(s_{a,\ell}, \sigma_a(s_{p,1}s_{a,1}\ldots s_{p,\ell}s_{a,\ell})) = s_{p,\ell+1}$, can be extended to a play. However, if $Plays^{\dot{\alpha}}$ is nonempty, we can transform $\dot{\mathcal{T}}^\alpha$ into a *non-blocking adviser restricted arena*

$$\mathcal{T}^\alpha = (S^\alpha, \langle S_p^\alpha, S_a^\alpha\rangle, s_{init}, U_p, U_a, T_p^\alpha \cup T_a^\alpha) \qquad (2)$$

that has the exact same set of plays $Plays^\alpha = Plays^{\dot{\alpha}}$ as $\dot{\mathcal{T}}^\alpha$ as outlined in Sec. II. Let us denote the sets of all protagonist's and adversary's strategies in $\mathcal{T}^\alpha$ by Σ_p^α and Σ_a^α, respectively. $Plays^{\sigma_p^\alpha, \Sigma_a^\alpha}$ refers to the set of plays induced by $\sigma_p^\alpha \in \Sigma_p^\alpha$ and Σ_a^α in $\mathcal{T}^\alpha$. If however $\dot{Plays}^\alpha$ is empty, a non-blocking adviser restricted arena $\mathcal{T}^\alpha$ does not exist.

Given the winning condition W_{Safe}, we define a good adviser α as one that permits the protagonist to achieve safety in the non-blocking adviser restricted arena $\mathcal{T}^\alpha$.

Definition 3 (Good adviser) *An adviser α is* good *for $(\mathcal{T}, W_{Safe})$ if there exists a non-blocking adviser restricted arena $\mathcal{T}^\alpha$ and a protagonist's strategy $\sigma_p^\alpha \in \Sigma_p^\alpha$, such that $Plays^{\sigma_p^\alpha, \Sigma_a^\alpha} \subseteq W_{Safe}$. Given a good adviser α, the set of protagonist's winning strategies is denoted by $\Omega_p^\alpha \subseteq \Sigma_p^\alpha$.*

Since there might be more good advisers, we need to distinguish which of them limit the adversary less and which of them more. To that end, we associate each adviser with a cost, called adviser level of limitation.

Definition 4 (Adviser level of limitation) *Given an arena $\mathcal{T}$ and a good adviser α, we define the* adviser level of limitation

$$\lambda(\alpha) = \inf_{\sigma_p^\alpha \in \Omega_p^\alpha} \gamma(\sigma_p^\alpha), \text{ where} \qquad (3)$$

$$\gamma(\sigma_p^\alpha) = \sup_{\sigma_a^\alpha \in \Sigma_a^\alpha} \limsup_{n \to \infty} \frac{1}{n} \sum_{j=1}^{n} \left|\alpha(\pi^{\sigma_p^\alpha, \sigma_a^\alpha}(2j))\right|. \qquad (4)$$

In other words, $\lambda(\alpha)$ is the *worst-case long-term average* of the number of forbidden inputs along the plays induced by the *best-case* protagonist's strategy σ_p^α. The choice of the worst-case long-term average is motivated by the fact that although the adversary can be advised, it cannot be controlled. On the other hand, the consideration of the best-case σ_p^α is due to the protagonist being fully controllable. We provide some intuitive explanations on the introduced terminology through the following illustrative example.

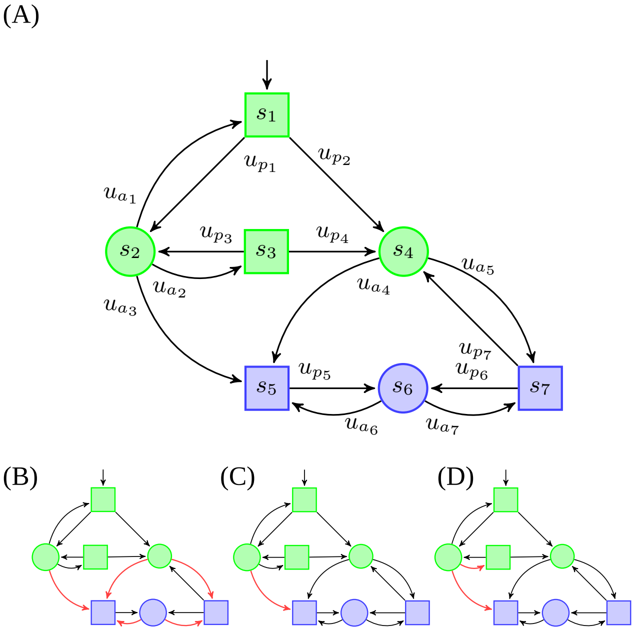

Fig. 1: (A) An example of a game arena with a safety winning condition. The protagonist's and adversary's states are illustrated as squares and circles, respectively. The safe set $Safe$ is in green, the unsafe set $Unsafe$ in blue. Transitions are depicted as arrows between them and they are labeled with the respective inputs that trigger them. (B) – (D) show three different advisers α_B, α_C and α_D, respectively, via marking the forbidden transitions in red.

Example 2 (Safety game and adviser) *An example of a game arena with a safety winning condition $W_{Safe}$is given in Fig. 1. (A). The squares illustrate the protagonist's states and the circles illustrate the adversary's ones. Transitions are depicted as arrows between them and they are labeled with the respective inputs that trigger them. The safe states in $Safe$ are shown in green and the unsafe ones in $Unsafe$ are in blue. Fig. 1.(B)-(D) show three advisers α_B, α_C and α_D, respectively, via marking the forbidden transitions in red. In Fig. 1.(B), $\alpha_B(s_2) = \{u_{a_3}\}$, $\alpha_B(s_4) = \{u_{a_4}, u_{a_5}\}$, and $\alpha_B(s_6) = \{u_{a_6}, u_{a_7}\}$. In Fig. 1.(C), $\alpha_C(s_2) = \{u_{a_3}\}$ and $\alpha_C(s_4) = \alpha_C(s_6) = \emptyset$. Finally, in Fig. 1.(D), $\alpha_D(s_2) = \{u_{a_2}, u_{a_3}\}$ and $\alpha_D(s_4) = \alpha_D(s_6) = \emptyset$.*

For α_B, the non-blocking adviser restricted arena contains states $S^{\alpha_B} = \{s_1, s_2, s_3\}$. The set of protagonist's strategies in $\mathcal{T}^{\alpha_B}$ is $\Sigma_p^{\alpha_B} = \{\sigma_p^{\alpha_B}\}$, such that $\sigma_p^{\alpha_B}(\pi(1)\ldots\pi(2j)s_1) = u_{p_1}$ and $\sigma_p^{\alpha_B}(\pi(1)\ldots\pi(2j)s_3) = u_{p_3}$, for all play prefixes $\pi(1)\ldots\pi(2j)$, $j \ge 0$ of all plays $\pi \in Plays^{\Sigma_p^{\alpha_B}, \Sigma_a^{\alpha_B}}$. Since $\sigma_p^{\alpha_B}$ is winning, α_B is good. It is easy to see that the set of protagonist's winning strategies and the set of all adversary's strategies in $\mathcal{T}^{\alpha_B}$ induce a set of plays $Plays^{\Omega_p^{\alpha_B}, \Sigma_a^{\alpha_B}} = \{s_1 s_2 \pi(3) s_2 \pi(5) s_2 \pi(7) s_2 \ldots \mid \pi(2j+1) \in \{s_1, s_3\}$, for all $j \ge 1\}$. The strategy $\sigma_p^{\alpha_B} \in \Omega_p^{\alpha_B}$ is therefore associated with the value $\gamma(\sigma_p^{\alpha_B}) = \sup_{\sigma_a^{\alpha_B} \in \Sigma_a^{\alpha_B}} \limsup_{n\to\infty} \frac{1}{n}\sum_{j=1}^{n} \left|\alpha_B(\pi^{\sigma_p^{\alpha_B}, \sigma_a^{\alpha_B}}(2j))\right| = \limsup_{n\to\infty} \frac{1}{n}\sum_{j=1}^{n} \left|\alpha_B(s_2)\right| = 1$, and the level of limitation of α_B is $\lambda(\alpha_B) = 1$. Although it might seem that adviser α_B is more limiting than α_C, it is not the case. The non-blocking adviser restricted arena $\mathcal{T}^{\alpha_C}$ in this case contains all states from $\mathcal{T}$, $S^{\alpha_C} = S$. However, the set of winning protagonist's strategies $\Omega_p^{\alpha_C}$ in $\mathcal{T}^{\alpha_C}$ is analogous as in case (B). Namely, if $\sigma_p^{\alpha_C}(\pi(1)\ldots\pi(2j)s_1) = u_{p_2}$ or $\sigma_p^{\alpha_C}(\pi(1)\pi(2)\ldots\pi(2j)s_3) = u_{p_4}$, the resulting play would not be winning for the protagonist as all ad-

versary's choices in s_4 lead to an unsafe state. Hence, $\mathit{Plays}^{\Omega_p^{\alpha_C},\Sigma_a^{\alpha_C}} = \mathit{Plays}^{\Omega_p^{\alpha_B},\Sigma_a^{\alpha_B}}$ and the level of limitation of α_C is $\lambda(\alpha_C) = 1$. Finally, α_D is more limiting than α_B and α_C. Following similar reasoning as above, we can see that $\mathit{Plays}^{\Omega_p^{\alpha_D},\Sigma_a^{\alpha_D}} = \{s_1 s_2 s_1 s_2 s_1 s_2 s_1 s_2 \ldots\}$, but since $|\alpha_D(s_2)| = 2$, we have $\lambda(\alpha_D) = \inf_{\sigma_p^{\alpha_D} \in \Omega_p^{\alpha_D}} \gamma(\sigma_p^{\alpha_D}) = \limsup_{n\to\infty} \frac{1}{n} \sum_{j=1}^{n} \big|\alpha_D(s_2)\big| = 2$.

Problem 1 *Consider $\mathcal{T} = (S, \langle S_p, S_a\rangle, s_{init}, U_p, U_a, T)$, and a safety winning condition W_{Safe} given via a partition $S = \langle \mathit{Safe}, \mathit{Unsafe}\rangle$. Synthesize an adviser $\alpha^\star$, and a protagonist's winning strategy $\sigma_p^{\alpha^\star\star}$, such that:*

- (i) *$\alpha^\star$ is good and $\sigma_p^{\alpha^\star\star} \in \Omega_p^{\alpha^\star}$,*
- (ii) *$\lambda(\alpha^\star) = \inf_{\alpha\in A} \lambda(\alpha)$, where A is the set of all good advisers for $(\mathcal{T}, W_{\mathit{Safe}})$, i.e. $\lambda(\alpha^\star)$ is least-limiting and*
- (iii) *$\gamma(\sigma_p^{\alpha^\star\star}) = \inf_{\sigma_p^{\alpha^\star} \in \Omega_p^{\alpha^\star}} \gamma(\sigma_p^{\alpha^\star})$, i.e. $\sigma_p^{\alpha^\star\star}$ is optimal.*

IV. Solution

Our solution builds on several steps: first, we generate a so-called *nominal adviser*, which assigns to each adversary state the set of forbidden inputs. We prove that the nominal adviser is by construction good, but does not have to be least-limiting. Second, building on the nominal adviser, we efficiently generate a finite set of candidate advisers. Third, the structural properties of the candidate advisers inherited from the properties of the nominal adviser allow us to prove that the problem of finding $\alpha^\star$ and $\sigma_p^{\alpha^\star\star}$ can be transformed to a mean-payoff game. By that, we prove that at least one $\sigma_p^{\alpha^\star\star}$ is memoryless and hence we establish decidability of Problem 1. Finally, we discuss how the set of the candidate advisers and their associated optimal protagonist's winning strategies can be used to guide an adversary who disobeys a subset of advises provided by a least-limiting adviser.

A. Nominal adviser

The algorithm to find the nominal adviser α^0 is summarized in Alg. 1. It systematically finds a set of states Losing, from which reaching of the unsafe set Unsafe cannot be avoided under any possible protagonist's and any adversary's choice of inputs. The set Losing is obtained via the computation of the finite converging sequence $\mathit{Unsafe} = \mathit{Losing}^0 \subset \mathit{Losing}^1 \subset \ldots \subset \mathit{Losing}^{n-1} = \mathit{Losing}^n = \mathit{Losing}$, $n \geq 0$, where for all $0 \leq j < n$, Losing^{j+1} is the set of states each of which either already belongs to Losing^j or has all outgoing transitions leading to Losing^j (line 15). The nominal adviser α^0 is set to forbid all transitions that lead to Losing (line 12). By construction, the algorithm terminates in at most $|S|$ iteration of the while loop (lines 9–16).

The following three lemmas summarize the key features of α^0 computed according to Alg. 1. The first two state that, if there exists a good adviser for $(\mathcal{T}, W_{\mathit{Safe}})$, then the nominal adviser is good. The third states that, if the nominal adviser forbids the adversary to apply an input $u_a \in \alpha^0(s_a)$ in a state s_a, then there does not exist a less limiting good adviser α', such that $u_a \notin \alpha'(s_a)$.

Algorithm 1: The nominal adviser α^0

```
   Data: $\mathcal{T} = (S, \langle S_p, S_a\rangle, s_{init}, U_p, U_a, T)$, and unsafe set
         $\mathit{Unsafe} \subseteq S$
   Result: $\alpha^0 : S_a \to 2^{U_a}$
1  forall the $s_a \in S_a$ do
2      $\alpha^0(s_a) := \emptyset$
3  end
4  forall the $s_a \in \mathit{Unsafe}$ do
5      $\alpha^0(s_a) := U_a^{s_a}$
6  end
7  $\mathit{Losing}^0 := \mathit{Unsafe}$
8  $j := 0$
9  while $j = 0$ or $\mathit{Losing}^j \neq \mathit{Losing}^{j-1}$ do
10     forall the $s_p \in \mathit{Losing}^j$ do
11         forall the $s_a, u_a$, such that $T(s_a, u_a) = s_p$ do
12             $\alpha^0(s_a) := \alpha^0(s_a) \cup \{u_a\}$
13         end
14     end
15     $\mathit{Losing}^{j+1} := \mathit{Losing}^j \cup \{s_i \in S_i \mid$
       $\bigcup_{u_i \in U_i^{s_i}} \{T_i(s_i, u_i)\} \subseteq \mathit{Losing}^j, i \in \{a, p\}\}$
16     $j := j + 1$
17 end
18 $\mathit{Losing} := \mathit{Losing}^j$
```

Lemma 1 *If $s_{init} \in \mathit{Losing}$ then there does not exist a good adviser for $(\mathcal{T}, W_{\mathit{Safe}})$.*

Proof: Suppose that $s_{init} \in \mathit{Losing}$ and there exist a good adviser α for $(\mathcal{T}, W_{\mathit{Safe}})$. Then there exists a non-blocking adviser restricted arena $\mathcal{T}^\alpha$ and a protagonist's strategy $\sigma_p^\alpha \in \Sigma_p^\alpha$, such that $\mathit{Plays}^{\sigma_p^\alpha,\Sigma_a^\alpha} \subseteq W_{\mathit{Safe}}$ in $\mathcal{T}^\alpha$. Consider a play $\pi = s_{p,1} s_{a,1} s_{p,2} s_{a,2} \ldots \in \mathit{Plays}^{\sigma_p^\alpha,\Sigma_a^\alpha}$ in $\mathcal{T}^\alpha$ and note that π does not intersect Unsafe. Suppose that $s_{p,1} = s_{init} \in \mathit{Losing} \setminus \mathit{Unsafe}$. Then there exists $j \geq 1$, such that $s_{init} \in \mathit{Losing}^j$, but $s_{init} \notin \mathit{Losing}^{j-1}$ and $\bigcup_{u_p \in U_p^{s_{init}}} \{T_p^\alpha(s_{init}, u_p)\} \subseteq \mathit{Losing}^{j-1}$ (line 15). Thus, $s_{a,1} \in \mathit{Losing}^{j-1}$. Furthermore, if $s_{a,1} \notin \mathit{Unsafe}$ then $\bigcup_{u_a \in U_a^{s_{a,1}}} \{T_a^\alpha(s_{a,1}, u_a)\} \subseteq \mathit{Losing}^{j-2}$ (line 15). Via inductive application of analogous arguments, we obtain that there exists $k \geq 1$, such that either $s_{p,k} \in \mathit{Losing}^0 = \mathit{Unsafe}$ or $s_{a,k} \in \mathit{Losing}^0 = \mathit{Unsafe}$. This contradicts the assumption that π is winning, i.e. the assumption that α is a good adviser. ■

Lemma 2 *If $s_{init} \notin \mathit{Losing}$, then α^0 computed by Alg. 1 is a good adviser.*

Proof: Let $s_{init} \notin \mathit{Losing}$. From the construction of Losing and α^0, it follows that for all $s_p \in S_p \setminus \mathit{Losing}$ there exists an input u_p and a state $s_a \in S_a \setminus \mathit{Losing}$, such that $T_a(s_p, u_p) = s_a$ (line 15). Furthermore, for all $s_a \in S_a \setminus \mathit{Losing}$ and all $u_a \in U_a^{s_a} \setminus \alpha^0(s_a)$ it holds that $T(s_a, u_a) \in S_p \setminus \mathit{Losing}$ (line 12). Hence, there exists a play $\pi \in \mathit{Plays}^{\dot{\alpha}^0}$ in $\dot{\mathcal{T}}^{\alpha^0}$, and thus there also exists a non-blocking adviser restricted arena $\mathcal{T}^{\alpha^0}$ and $\sigma_p^{\alpha^0} \in \Sigma_p^{\alpha^0}$, such that any play $\pi \in \mathit{Plays}^{\sigma_p^{\alpha^0},\Sigma_a^{\alpha^0}}$ in $\mathcal{T}^{\alpha^0}$ does not intersect Losing. Because $\mathit{Unsafe} \subseteq \mathit{Losing}$, it holds that $\mathit{Plays}^{\sigma_p^{\alpha^0},\Sigma_a^{\alpha^0}} \subseteq W_{\mathit{Safe}}$ and adviser α^0 is good. ■

Intuitively, Lemmas 1 and 2 state that the restrictions imposed by the nominal adviser α^0 were sufficient. As a corollary, it also holds that the non-blocking nominal adviser restricted arena $\mathcal{T}^{\alpha^0}$ does not contain any state in Losing and therefore that all plays in $\mathcal{T}^{\alpha^0}$ are winning. Note however, that the nominal adviser does not have to be least-limiting. As we illustrate through the following example, imposing additional restrictions on the adversary's choices might, perhaps surprisingly, lead to the avoidance of adversary's states, where a high number of inputs are forbidden.

Example 3 *An example of a safety game is shown in Fig. 2.(A). The result of the nominal adviser computation according to Alg. 1 is illustrated in Fig. 2.(B). Namely,* $\mathit{Losing} = \{s_4\}$, *and* $\alpha^0(s_2) = \{u_{a_2}\}$, $\alpha^0(s_6) = \{u_{a_5}\}$, *and* $\alpha^0(s_7) = \emptyset$. *There is only one protagonist's strategy* $\{\sigma_p^{\alpha^0}\} = \Sigma_p^{\alpha^0}$, *and it is winning* $\sigma_p^{\alpha^0} \in \Omega_p^{\alpha^0}$ *since* $\mathit{Plays}^{\sigma_p^{\alpha^0}, \Sigma_a^{\alpha^0}} = \{s_1 s_2 s_3 s_6 s_3 s_6 \ldots, s_1 s_2 s_5 s_7 s_5 s_7 \ldots\}$. *The level of limitation of* α^0 *is thus* $\lambda(\alpha^0) = \sup_{\sigma_a^\alpha \in \Sigma_a^\alpha} \limsup_{n\to\infty} \frac{1}{n} \sum_{j=1}^n \left|\alpha(\pi^{\sigma_p^\alpha, \sigma_a^\alpha}(2j))\right| = \limsup_{n\to\infty} \frac{1}{n}\Big(\left|\alpha(s_2)\right| + \sum_{j=2}^n \left|\alpha(s_3)\right|\Big) = 1$. *Loosely speaking, the worst-case adversary's strategy* σ_a *that respects the nominal adviser* α^0 *takes the play to the left-hand branch of the system.*

Fig. 2.(C) shows an alternative adviser α' *that guides each play to the right-hand branch of the system. It is good since there is a non-blocking adviser limited arena* $\mathcal{T}^{\alpha'}$ *and the only protagonist's strategy* $\sigma_p^{\alpha'} \in \Sigma_p^{\alpha'}$ *on* $\mathcal{T}^{\alpha'}$ *is winning, since* $\mathit{Plays}^{\sigma_p^{\alpha'}, \Sigma_a^{\alpha'}} = \{s_1 s_2 s_5 s_7 s_5 s_7 \ldots\}$. *The level of limitation of* α' *is* $\lambda(\alpha') = \limsup_{n\to\infty} \frac{1}{n}\Big(\left|\alpha(s_1)\right| + \sum_{j=2}^n \left|\alpha(s_5)\right|\Big) = \limsup_{n\to\infty} \frac{1}{n}\Big(\left|\alpha(s_1)\right|\Big) \ll 1$. *Hence,* α' *is less limiting than the nominal adviser* α^0.

Lemma 3 *Consider an adviser* α' *for* $(\mathcal{T}, W_{\mathit{Safe}})$ *and suppose that there exists a state* $s_a \in S_a$ *and* $u_a \in U_a$, *such that* $u_a \in \alpha^0(s_a)$ *and* $u_a \notin \alpha'(s_a)$. *Then* α' *is either not good or at least as limiting as the nominal adviser* α^0, *i.e.* $\lambda(\alpha^0) \leq \lambda(\alpha')$.

Proof: The proof is lead by contradiction. Consider an adviser α' for $(\mathcal{T}, W_{\mathit{Safe}})$. Suppose that there exists a state $s_a \in S_a$ and $u_a \in U_a$, such that $u_a \in \alpha^0(s_a)$ and $u_a \notin \alpha'(s_a)$ and α' is good. Furthermore, let $\Omega_p^{\alpha'}$ be the set of protagonist's winning strategies on the non-blocking adviser restricted arena $\mathcal{T}^{\alpha'}$. and assume that α' is less limiting that α^0, i.e. that $\lambda(\alpha') < \lambda(\alpha^0)$. Then from the definition of λ in Eq. (3), there exists a protagonist's strategy $\sigma_p^{\alpha'} \in \Omega_p^{\alpha'}$, such that $\gamma(\sigma_p^{\alpha'}) < \lambda(\alpha^0)$. Henceforth, there also exists a winning play $\pi = s_{p,1} s_{a,1} s_{p,2} s_{a,2} \ldots \in \mathit{Plays}^{\sigma_p^{\alpha'}, \Sigma_a^{\alpha'}}$ on $\mathcal{T}^{\alpha'}$ with the property that for some $k \geq 1$, $s_{p,k+1} \in T(s_{a,k}, u_a)$, where $u_a \notin \alpha'(s_{a,k})$ and $u_a \in \alpha^0(s_{a,k})$. If such a winning play does not exist, it holds that $\gamma(\sigma_p^{\alpha'}) \geq \lambda(\alpha^0)$, which contradicts the assumption that α' is less limiting than α^0. Since $u_a \in \alpha^0(s_{a,k})$, it holds $s_{p,k+1} \in \mathit{Losing}$ by construction (line 12). Either $s_{p,k+1} \in \mathit{Unsafe}$, which directly contradicts the assumption that α' is good, or $s_{p,k+1} \in \mathit{Losing}^j$, for some $j \geq 1$. From the iterative construction of Losing, we obtain $s_{a,k+1} \in \mathit{Losing}^{j-1}$ and if $s_{a,k+1} \notin \mathit{Unsafe}$, then $s_{p,k+2} \in \mathit{Losing}^{j-2}$. By inductive reasoning it follows that there exists $\ell \geq k+1$, such that either $s_{p,\ell} \in \mathit{Losing}^0 = \mathit{Unsafe}$, or $s_{a,\ell} \in \mathit{Losing}^0 = \mathit{Unsafe}$. This contradicts the assumption that π is winning, i.e. the assumption that α' is good. ■

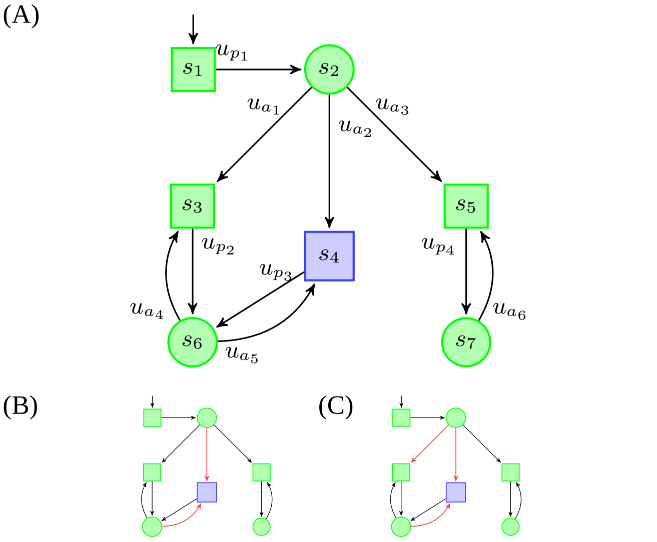

Fig. 2: (A) An example of a game arena with a safety winning condition. The protagonist's and adversary's states are illustrated as squares and circles, respectively. The safe set Safe is in green, the unsafe set Unsafe in blue. Transitions are depicted as arrows between them and they are labeled with the respective inputs that trigger them. (B) shows the nominal adviser α^0 and Losing via marking the forbidden transitions and the states in Losing in red. (C) shows an alternative adviser α' that is also good and less limiting than α^0.

Thanks to Lemma 3, we know that there exists a good adviser $\alpha^\star$ that is least-limiting and builds on the nominal one in the following sense: $\alpha^0(s_a) \subseteq \alpha^\star(s_a)$, for all $s_a \in S_a$. Whereas following the nominal adviser is essential for maintaining the system safety, following the additional restrictions suggested by $\alpha^\star$ can be perceived as a weak form of advice. If this advice is not respected by the adversary, safety is not necessarily going to be violated, however, in order to maintain safety, the adversary might need to obey further, more limiting advises. We will discuss later on in Sec. IV-D how to use both the combination of a least-limiting adviser and the nominal one in order to guide the adversary during the system execution (the play on the game arena).

B. Least-limiting solution

Let $\dot{A}_{cand}$ denote the finite set of candidate advisers obtained from the nominal adviser α^0, $\dot{A}_{cand} = \{\alpha \mid \alpha^0(s_a) \subseteq \alpha(s_a)$, for all $s_a \in S_a\}$. Note that $\alpha \in \dot{A}_{cand}$ does not have to be good since it might not allow for an existence of a non-blocking adviser restricted arena $\mathcal{T}^\alpha$. As outlined in Sec. II, it can be however decided whether $\dot{\mathcal{T}}^\alpha$ from Def. 2 has an equivalent non-blocking arena $\mathcal{T}^\alpha$. Building on ideas from Lemmas 1 and 2, we can easily see that the existence of non-blocking adviser restricted arena

$\mathcal{T}^{\alpha}$ also implies the existence of a protagonist's winning strategy $\sigma_p^{\alpha} \in \Omega_p^{\alpha}$. In fact, because states from $Losing$ were removed from $\mathcal{T}^{\alpha^0}$ (lines 4–6, 9–16 of Alg. 1), all plays in $\mathcal{T}^{\alpha}$ are winning and $\Sigma_p^{\alpha} = \Omega_p^{\alpha}$.

$$A_{cand} = \{\alpha \in \dot{A}_{cand} \mid \alpha \text{ is a good adviser}\}. \tag{5}$$

From Lemma 3 and the construction of A_{cand}, at least one least-limiting good adviser belongs to A_{cand}. In the remainder of the solution, we focus on solving the following sub-problem for each $\alpha \in A_{cand}$.

Problem 2 *Consider a good adviser $\alpha \in A_{cand}$. Find $\lambda(\alpha)$ and an optimal protagonist's winning strategy $\sigma_p^{\alpha\star}$ with $\gamma(\sigma_p^{\alpha\star}) = \inf_{\sigma_p^{\alpha} \in \Omega_p^{\alpha}} \gamma(\sigma_p^{\alpha}) = \inf_{\sigma_p^{\alpha} \in \Sigma_p^{\alpha}} \gamma(\sigma_p^{\alpha})$.*

We propose to translate Problem 2 to finding an optimal strategy to a mean-payoff game on a modified arena $\widetilde{\mathcal{T}}^{\alpha}$:

Definition 5 (Mean-payoff game arena $\widetilde{\mathcal{T}}^{\alpha}$) *Given a non-blocking adviser restricted arena $\mathcal{T}^{\alpha} = (S^{\alpha}, \langle S_p^{\alpha}, S_a^{\alpha}\rangle, s_{init}, U_p, U_a, T_p^{\alpha} \cup T_a^{\alpha})$, we define the mean-payoff game arena $\widetilde{\mathcal{T}}^{\alpha} = (\mathcal{T}^{\alpha}, w)$, where for all $\widetilde{T}_p(s_p, u_p) = s_a$, $w(s_p, s_a) = -|\alpha(s_a)|$ and for all $\widetilde{T}_a(s_a, u_a) = s_p$, $w(s_a, s_p) = 0$.*

Lemma 4 *Problem 2 reduces to the problem of optimal strategy synthesis for the mean-payoff game $\widetilde{\mathcal{T}}^{\alpha}$.*

Proof: The optimal strategy $\widetilde{\sigma}_p^{\alpha\star}$ for the mean-payoff game $\widetilde{\mathcal{T}^{\alpha}}$ obtained e.g., by the algorithm from [5] has the value $\nu(\widetilde{\sigma}_p^{\alpha\star}) =$

$$\sup_{\sigma_p^{\alpha} \in \Sigma_p^{\alpha}} \inf_{\sigma_a^{\alpha} \in \Sigma_a^{\alpha}} \liminf_{n \to \infty} \frac{1}{n} \sum_{j=1}^{n} w(\pi^{\sigma_p^{\alpha}, \sigma_a^{\alpha}}(j), \pi^{\sigma_p^{\alpha}, \sigma_a^{\alpha}}(j+1)) =$$

$$\inf_{\sigma_p^{\alpha} \in \Sigma_p^{\alpha}} \sup_{\sigma_a^{\alpha} \in \Sigma_a^{\alpha}} \limsup_{n \to \infty} \frac{1}{n} \sum_{j=1}^{n} -w(\pi^{\sigma_p^{\alpha}, \sigma_a^{\alpha}}(j), \pi^{\sigma_p^{\alpha}, \sigma_a^{\alpha}}(j+1)) =$$

$$\inf_{\sigma_p^{\alpha} \in \Sigma_p^{\alpha}} \sup_{\sigma_a^{\alpha} \in \Sigma_a} \limsup_{n \to \infty} \frac{1}{n} \sum_{j=1}^{n} |\alpha(\pi^{\sigma_p^{\alpha}, \sigma_a^{\alpha}}(2j)| = \lambda(\alpha).$$

Furthermore, as noted above $\Sigma_p^{\alpha} = \Omega_p^{\alpha}$ and hence the proof is complete. ■

It has been shown in [8] that in mean-payoff games, memoryless strategies suffice to achieve the optimal value. In fact, using the algorithm from [5], the strategy $\widetilde{\sigma}_p^{\alpha\star}$ takes the form of a memoryless strategy $\widetilde{\varsigma}_p^{\alpha\star} : S_p^{\alpha} \to U_p$.

C. Overall solution

We summarize how the algorithms from Sec. IV-A and Sec. IV-B serve in finding a solution to Problem 1. 1) The nominal adviser α^0 is built according to Alg. 1. If there does not exist a non-blocking adviser restricted arena $\mathcal{T}^{\alpha^0}$, then there does not exist a solution to Problem 1. 2) The set of candidate advisers A_{cand} is built according to Eq. (5). 3) For each candidate adviser $\alpha \in A_{cand}$, the value $\lambda(\alpha)$ and the memoryless optimal protagonist's winning strategy $\varsigma_p^{\alpha\star} \in \Omega_p^{\alpha}$ are computed through the translation to a mean-payoff game optimal strategy synthesis according to Def. 5. 4) An adviser $\alpha^\star \in A_{cand}$ with $\lambda(\alpha^\star) = \inf_{\alpha \in A_{cand}} \lambda(\alpha)$ together with its associated optimal strategy $\varsigma_p^{\alpha^\star\star}$ are the solution to Problem 1.

D. Guided system execution

Finally, we discuss how the set of good advisers A_{cand} can be used to guide the adversary on-the-fly during the system execution. Given an adviser $\alpha \in A_{cand}$, let us call the fact that $u_a \in \alpha(s_a)$ an *advise*. We distinguish two types of advises, *hard* and *soft*. Hard advises are the ones imposed by the nominal adviser, $u_a \in \alpha^0(s_a)$, while soft are the remaining ones that can be violated without jeopardizing the system safety. The goal of the guided execution is to permit the adversary to disobey a soft advise and react to this event via a switch to another, possibly more limiting adviser that does not contain this soft advise. Let $\preceq$ be a partial ordering on the set A_{cand}, where $\alpha \preceq \alpha'$ if $\alpha(s_a) \subseteq \alpha'(s_a)$, for all $s_a \in S_a$. Hence, for the nominal adviser α^0, it holds that $\alpha^0 \preceq \alpha$, for all $\alpha \in A_{cand}$.

The system execution that corresponds to a play in $\mathcal{T}$ proceeds as follows: 1) The system starts at the initial state $s_{curr} = s_{init}$ with the current adviser being least-limiting adviser $\alpha_{curr} = \alpha^\star$ and the current protagonist's strategy being the memoryless winning strategy $\varsigma_{p,curr} = \varsigma_p^{\alpha^\star\star}$. 2) The input $\varsigma_{p,curr}(s_{curr})$ is applied by the protagonist and the system changes its current state s_{curr} according to T_p. The current state belongs to the adversary. 3) $\alpha_{curr}(s_{curr})$ is provided. The adversary chooses an input $u_a \in U_a^{s_{curr}}$. a) If $u_a \notin \alpha_{curr}(s_{curr})$, then the system updates its state s_{curr} according to T_a and proceeds with step 2. b) If $u_a \in \alpha^0(s_{curr})$ then hard advise is disobeyed and system safety will be unavoidably violated and the system needs to stop immediately. c) If $u_a \in \alpha_{curr}(s_{curr})$, but $u_a \notin \alpha^0(s_{curr})$, then only a soft advise is disobeyed. The current adviser α_{curr} is updated to α', with the property that $\lambda(\alpha') = \inf_{\alpha \in A_{\preceq}} \lambda(\alpha)$, where $A_{\preceq} = \{\alpha \in A_{cand} \mid \alpha \preceq \alpha_{curr}\}$ and the current protagonist's strategy $\varsigma_{p,curr}$ is updated to $\varsigma_p^{\alpha'\star}$. The current state s_{curr} is updated according to T_a and the system proceeds with step 2).

Example 4 *Consider the safety game in Fig. 3.(A). The result of the nominal adviser computation according to Alg. 1 is illustrated in Fig. 3.(B). Namely, $\alpha^0(s_2) = \{u_{a_2}\}$, $\alpha^0(s_6) = \emptyset$, $\alpha^0(s_7) = \{u_{a_5}\}$, $\alpha^0(s_8) = \{u_{a_7}, u_{a_8}\}$, and $\alpha^0(s_{11}) = u_{a_9}$. The states in $Losing$ are marked in red. Fig. 3.(C) shows the non-blocking adviser restricted arena $\mathcal{T}^{\alpha^0}$ with the removed states and transitions in light grey. The corresponding optimal protagonist's winning strategy $\varsigma_p^{\alpha^0\star}$ in $\mathcal{T}^{\alpha^0}$ is highlighted in green in Fig. 3.(B), i.e. $\varsigma_p^{\alpha^0\star}(s_1) = u_{p_1}$, $\varsigma_p^{\alpha^0\star}(s_3) = u_{p_2}$, $\varsigma_p^{\alpha^0\star}(s_5) = u_{p_5}$, and $\varsigma_p^{\alpha^0\star}(s_9) = u_{p_6}$. The level of limitation of α^0 is $\lambda(\alpha^0) = \limsup_{n\to\infty} \frac{1}{n}(|\alpha^0(s_2)| + \sum_{j=2}^{n} |\alpha^0(s_8)|) = \limsup_{n\to\infty} \frac{1}{n}(2n-1)$. Fig. 3.(D) shows least-limiting adviser $\alpha^\star$. As opposed to α^0, $\alpha^\star(s_2) = \{u_{a_2}, u_{a_3}\}$, where the advise $u_{a_3} \in \alpha^\star(s_2)$ (in magenta) is soft. Fig. 3.(E) illustrates the non-blocking adviser restricted arena $\mathcal{T}^{\alpha^\star}$. The optimal protagonist's winning strategy is the only protagonist's strategy in $\mathcal{T}^{\alpha^\star}$. The level of limitation of $\alpha^\star$ is $\lambda(\alpha^\star) = \limsup_{n\to\infty} \frac{1}{n}(|\alpha^\star(s_2)| + \sum_{j=2}^{n} |\alpha^0(s_6)|)$ $= \limsup_{n\to\infty} \frac{1}{n} < \lambda(\alpha^0)$. There exist more good advisers $\alpha' \in A_{cand}$. For each of them, either $\lambda(\alpha') = \lambda(\alpha^0)$ or $\lambda(\alpha') = \lambda(\alpha^\star)$.*

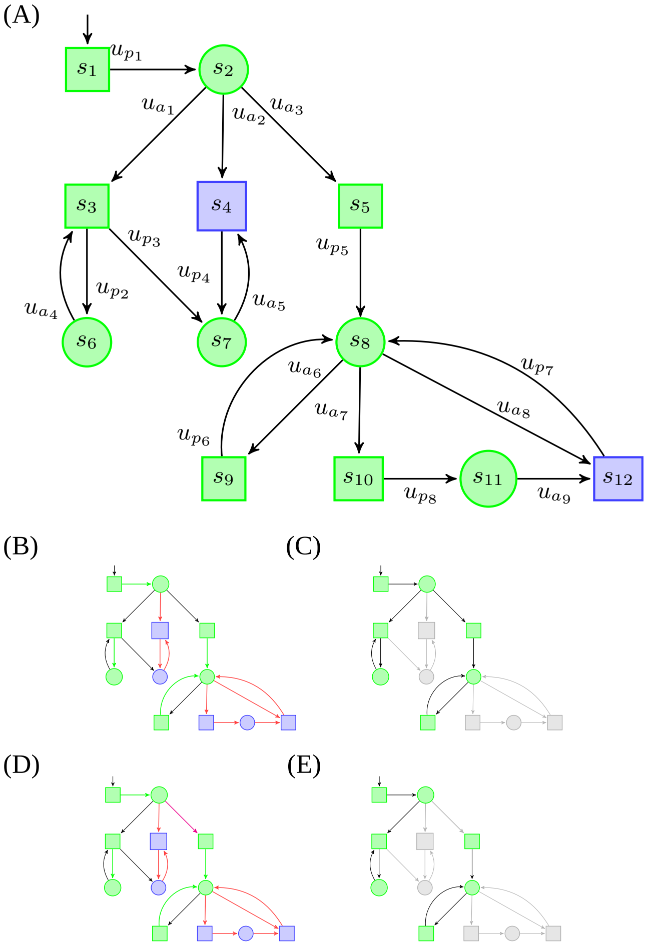

Fig. 3: (A) An example of a game arena with a safety winning condition. (B) The nominal adviser α^0 and *Losing* via marking the forbidden transitions and states in *Losing* in red. $\varsigma_p^{\alpha^0\star}$ is in green. (C) The non-blocking adviser restricted arena $\mathcal{T}^{\alpha^0}$. (D) $\alpha^\star$ and (E) The non-blocking adviser restricted arena $\mathcal{T}^{\alpha^\star}$.

The guided system execution proceeds as follows: The system starts in state $s_{curr} = s_{p_1}$ *with* $\alpha_{curr} = \alpha^\star$ *and* $\varsigma_{p,curr} = \varsigma_p^{\alpha^\star\star}$. *Input* u_{p_1} *is applied,* $s_{curr} = s_2$. *Then,* $\alpha_{curr}(s_{curr}) = \alpha^\star(s_2)$ *is provided. The adversary chooses either* u_{a_1}, u_{a_2}, *or* u_{a_3}, *but, through the adviser it is recommended not to select* u_{a_3} *(soft advise) and* u_{a_2} *(hard advise). If the choice is* u_{a_1}, *the system state is updated to* $s_{curr} = s_3$, *and in the remainder of the execution, the protagonist and the adversary apply* u_{p_2} *and* u_{a_4}, *respectively, switching between states* s_3 *and* s_6. *If the choice is* u_{a_3}, *a soft advice is disobeyed, the current state becomes* s_5 *and the current adviser and strategy are updated to* $\alpha_{curr} = \alpha^0$ *and* $\varsigma_{p,curr} = \varsigma_p^{\alpha^0\star}$, *which satisfy that* $\lambda(\alpha^0) = \inf_{\alpha\in\mathcal{A}_{\preceq}} \lambda(\alpha)$. *Input* u_{p_5} *is then applied and* $s_{curr} = s_8$. *In the remainder of the execution, the adversary is guided to follow the hard advices* $u_{a_7}, u_{a_8} \in \alpha_{curr}(s_8)$, *leading the system to switching between* s_8 *and* s_9. *If the choice in* s_2 *is* u_{a_2} *despite the hard advice, the system reaches an unsafe state.*

V. Conclusions and Future Work

We have studied the problem of synthesizing least-limiting guidelines for decision making in semi-autonomous systems involving entities that are uncontrollable, but partially willing to collaborate on achieving safety of the overall system. We have proposed a rigorous formulation of such problem and an algorithm to synthesize least-limiting advisers for an adversary in a 2-player safety game and we have proposed a systematic way to guide the system execution with their use. As far as we are concerned, this paper presents one of the first steps towards studying the problem of synthesizing guidelines for uncontrollable entities. Future work naturally includes extensions to more complex winning conditions, different measures of level of violation, and continuous state spaces. We also plan to implement the algorithms and show their potential in a case study.

References

[1] R. Alur, S. Moarref, and U. Topcu. Counter-strategy guided refinement of GR(1) temporal logic specifications. In *Formal Methods in Computer-Aided Design*, pages 26–33. IEEE, 2013.

[2] K. R. Apt and E. Grädel, editors. *Lectures in Game Theory for Computer Scientists*. Cambridge University Press, 2011.

[3] E. Bartocci, R. Grosu, P. Katsaros, C. Ramakrishnan, and S. Smolka. Model repair for probabilistic systems. In *Tools and Algorithms for the Construction and Analysis of Systems*, volume 6605 of *LNCS*, pages 326–340. Springer Berlin Heidelberg, 2011.

[4] J. D. W. I. Bernet, Julien. Permissive strategies : from parity games to safety games. *Theoretical Informatics and Applications*, 36(3):261–275, 2002.

[5] L. Brim, J. Chaloupka, L. Doyen, R. Gentilini, and J. Raskin. Faster algorithms for mean-payoff games. *Formal Methods in System Design*, 38(2):97–118, 2011.

[6] K. Chatterjee, T. Henzinger, and B. Jobstmann. Environment assumptions for synthesis. In *Concurrency Theory (CONCUR)*, volume 5201 of *LNCS*, pages 147–161. Springer Berlin Heidelberg, 2008.

[7] T. Chen, E. M. Hahn, T. Han, M. Kwiatkowska, H. Qu, and L. Zhang. Model repair for Markov decision processes. In *International Symposium on Theoretical Aspects of Software Engineering (TASE)*, pages 85–92. IEEE, 2013.

[8] A. Ehrenfeucht and J. Mycielski. Positional strategies for mean payoff games. *International Journal of Game Theory*, 8(2):109–113, 1979.

[9] M. Faella. Best-effort strategies for losing states. *CoRR*, abs/0811.1664, 2008.

[10] S. Hirche and M. Buss. Human-oriented control for haptic teleoperation. *Proceedings of the IEEE*, 100(3):623–647, 2012.

[11] M. Kloetzer and C. Belta. Dealing with nondeterminism in symbolic control. In *Hybrid Systems: Computation and Control (HSCC)*, pages 287–300, Berlin, Heidelberg, 2008. Springer-Verlag.

[12] H. Kress-Gazit, G. E. Fainekos, and G. J. Pappas. Temporal logic-based reactive mission and motion planning. *IEEE Transactions on Robotics*, 25(6):1370–1381, 2009.

[13] W. Li, L. Dworkin, and S. A. Seshia. Mining assumptions for synthesis. In *ACM/IEEE International Conference on Formal Methods and Models for Codesign (MEMOCODE)*, 2011.

[14] M. Mazo and M. Cao. Design of reward structures for sequential decision-making processes using symbolic analysis. In *American Control Conference (ACC), 2013*, pages 4393–4398, 2013.

[15] K. Savla, T. Temple, and E. Frazzoli. Human-in-the-loop vehicle routing policies for dynamic environments. In *IEEE Conference on Decision and Control (CDC)*, pages 1145–1150, 2008.

[16] J. Tumova, G. C. Hall, S. Karaman, E. Frazzoli, and D. Rus. Least-violating control strategy synthesis with safety rules. In *Hybrid Systems: Computation and Control (HSCC)*, pages 1–10. ACM, 2013.

[17] A. van Hulst, M. Reniers, and W. Fokkink. Maximally permissive controlled system synthesis for modal logic. In *Theory and Practice of Computer Science (SOFSEM)*, volume 8939 of *LNCS*, pages 230–241. Springer Berlin Heidelberg, 2015.

[18] T. Wongpiromsarn, U. Topcu, and R. M. Murray. Receding horizon control for temporal logic specifications. In *Hybrid Systems: Computation and Control (HSCC)*, pages 101–110, 2010.