Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)
Survey Track

Goal Recognition Design - Survey

Sarah Keren[1*], **Avigdor Gal**[2] and **Erez Karpas**[2]
[1]School of Engineering and Applied Sciences, Harvard University
[2]Technion – Israel Institute of Technology
skeren@seas.harvard.edu, {avigal,karpase}@technion.ac.il

Abstract

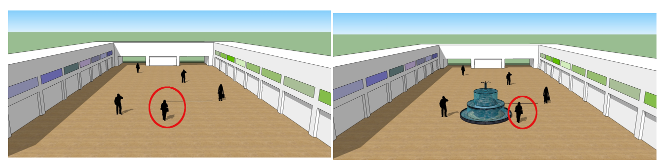
Goal recognition is the task of recognizing the objective of agents based on online observations of their behavior. *Goal recognition design* (*GRD*), the focus of this survey, facilitates goal recognition by the analysis and redesign of goal recognition models. In a nutshell, given a model of a domain and a set of possible goals, a solution to a *GRD* problem determines: (1) to what extent do actions performed by an agent reveal the agent's objective? and (2) what is the best way to modify the model so that the objective of an agent can be detected as early as possible? *GRD* answers these questions by offering a solution for assessing and minimizing the maximal progress of any agent before recognition is guaranteed. This approach is relevant to any domain in which efficient goal recognition is essential and in which the model can be redesigned. Applications include intrusion detection, assisted cognition, computer games, and human-robot collaboration. This survey presents the solutions developed for evaluation and optimization in the *GRD* context, a discussion on the use of *GRD* in a variety of real-world applications, and suggestions of possible future avenues of *GRD* research.

1 Introduction and Overview

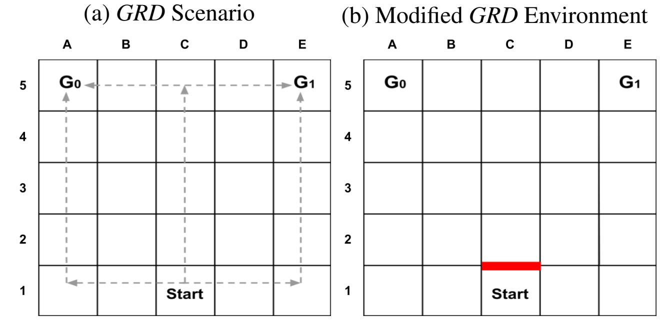
Goal recognition design (*GRD*) involves the analysis and redesign of goal recognition settings in order to enhance the ability to recognize the goals of agents that are operating in some environment. As such, *GRD* extends the task of goal recognition [Kautz, 1987; Carberry, 2001; Ramirez and Geffner, 2010; Sukthankar *et al.*, 2014; Vered and Kaminka, 2017; Pereira *et al.*, 2017], which aims at discovering the goals of agents based on observations of their behavior collected online, by allowing design changes to the goal recognition setting.

This line of research is motivated by applications in various domains, from urban transportation to medical informatics, which require systems to automatically and rapidly analyze agents' behavior, identify their goals, and then help them reach (or, depending on the application, prevent them from reaching) those goals. Potential applications include intrusion detection [Jarvis *et al.*, 2004], computer games [Ha *et al.*, 2011], and human-robot collaboration [Levine and Williams, 2018; Freedman and Zilberstein, 2017; Albrecht and Stone, 2018]. Such applications make use of recent developments in data science, which offers efficient and effective tools for gathering, managing, analyzing, and visualizing big amounts of data in order to observe and predict the intentions of agents. *GRD* supports rapid goal recognition whenever the model design of the application can be controlled.

A *GRD* problem includes a description of a goal recognition setting with a set of possible goals and a description of the available ways to modify the system. The *GRD* analysis consists of two main stages, namely evaluation and optimization. In the evaluation stage, we assess to what extent the goal of an agent acting in the system may remain unrecognized, while the optimization stage consists of finding the best way to modify the system so as to facilitate recognition.



(a) *GRD* Scenario (b) Modified *GRD* Environment



(c) *GRD* Model (d) Modified *GRD* Model

Figure 1: *GRD* example

Example 1 *A simple* GRD *problem is presented in Figure 1, positioned in the context of airport security. The model consists of a simple room with a single entry point ('Start') and*

[*]Contact Author

two possible exit points, marked as 'G_0' (boarding gates for domestic flights) and 'G_1' (international flights). Agents can move vertically or horizontally from 'Start' to either of these goals (see Figure 1c). Assuming agents behave optimally and are unaware or agnostic to their behavior being monitored, for each goal there are several possible paths (a subset of possible paths is marked by dashed lines in Figure 1c). As illustrated, paths to different goals may share a common prefix. In this model, an agent's goal becomes clear once turning left or right.

GRD introduces measures of model quality in terms of goal recognition. In this example, *worst case distinctiveness* (*WCD*) examines how long an agent can operate in a model without its[1] goal being identified by an observer; the longer it takes to reveal the agent's goal, the worse is the model. In Figure 1a, for example, the circled user can walk all the way up to the opposite side of the terminal (four steps in Figure 1c) before revealing her true intention.

Once the quality of a model is defined, it can be improved. Towards this end, means of model modification are defined. An example of a change to a model is through restrictions on the set of available actions an agent can perform. To maintain user comfort, a *GRD* solution may be required to preserve the original solution cost for all goals. In addition, we may wish to attain maximal achievable improvement while minimizing introduced changes and respecting any design constraints that may be specified. To illustrate, assume now that airport managers can place barriers in the terminal to control the flow of passengers (a common and effective solution for passenger control), but also wish to minimize obstruction to the ease of use of the terminal. Figures 1b and 1d present a solution to a *GRD* problem, where a single modification (adding a fountain at the room entrance) reduces *WCD* from 4 to 0 without increasing the minimal cost to any of the goals.

The ideas presented above can be applied to a variety of goal recognition settings. For example, consider a smart home in which the activity of a user with a physical or mental disability is tracked so as to help the user perform daily activities and avoid hazards. In such a setting, the environment may need to be redesigned (e.g., by furniture repositioning) so that a goal recognition system can detect dangerous situations as early as possible (e.g., approaching a hot oven).

The initial work on *GRD* [Keren *et al.*, 2014] (and the example above) puts forward a model based on three simplifying assumptions, namely that the environment is fully observable both to the goal recognition system and to the acting agents; that the outcomes of agent actions are deterministic; and that agents, who are agnostic to the goal recognition system, act optimally. The model is modified by disallowing (removing) actions from the set of applicable actions. Several extensions have since been suggested to this basic setting. The objective of this survey is to describe the components that comprise a *GRD* problem, against which we describe the different extensions that have been suggested in the literature since the first *GRD* setting was introduced.

GRD, with its emphasis on analysis and redesign of goal recognition models, is closely related to and complements goal recognition research efforts, where the objective is to recognize goals (or plans for plan recognition) of agents on the basis of their observed behavior [Sukthankar *et al.*, 2014]. Commonalities, as well as differences, have been highlighted in multiple discussions at the PAIR (Plan, Activity, and Intent Recognition) workshop series.[2] However related, *GRD* is a different task. While goal recognition aims at discovering the goals of an agent by analyzing a specific observation sequence, *GRD* analyzes the goal recognition setting, offering a solution to facilitate online recognition by allowing early detection of an agent's goal.

GRD is also related and complements a variety of recent lines of work on explainable and privacy preserving AI that consider settings where observers aim at recognizing the goals and plans of agents, while agents can choose to behave in a way that either explains or obfuscates their intentions [MacNally *et al.*, 2018; Chakraborti *et al.*, 2017; Chakraborti *et al.*, 2019; Kulkarni *et al.*, 2019b]. The *GRD* framework suggests a third and possibly complementary approach, where the ability to perform goal recognition is controlled by the design of the goal recognition setting.

The specific use of *GRD* tools depends on the analyzed setting and the relationship between the agents within it. If acting agents (actors) are agnostic or unaware of the recognition process, as in our depicted example, their behavior is unaffected by the observer's presence. In adversarial settings, *GRD* can be used to support early goal recognition of attackers by an observing agent or goal recognition system [Kabanza *et al.*, 2010; Keren *et al.*, 2015; Masters and Sardina, 2017]. In addition, Keren *et al.* (2016b) show that privacy-preserving actors can use *GRD* tools to identify plans that lead to their destination while keeping their goal ambiguous as long as possible. Even in collaborative settings, minimizing *WCD* guarantees a bound on the number of observations that need to be collected before recognition is achieved, thus reducing the need to generate explanations or incur the costs of performing expensive behaviors that align with the observer's expectations [MacNally *et al.*, 2018; Kulkarni *et al.*, 2019a].

In the rest of this survey, we provide a description of the main components of a *GRD* model (Section 2), and then dive into the *GRD* extensions that have been developed to account for different variations of the three components, namely environment dynamics (Section 3), actor types (Section 4), and observer's sensor models (Section 5). We conclude with a discussion of approaches to (re)design (Section 6) and outlook on future directions and challenges (Section 7). Table 1 lists the *GRD* models presented so far in the literature.

2 Components of a GRD Problem

A *GRD* problem has two main components; the analyzed goal recognition setting (Section 2.1), and a design model specifying possible ways to modify the goal recognition setting (Section 2.2).

[1] Agents are referred to as she, he, and it interchangeably, depending on the context.

[2] planrec.org

2.1 Goal Recognition

There are multiple ways to define goal recognition in the literature [Carberry, 2001; Sukthankar *et al.*, 2014; Ramirez and Geffner, 2010; Vered and Kaminka, 2017]. This task typically includes the analysis of a specific observation sequence to be mapped to a set of possible goals the agent may be trying to achieve. This is one of the major differences from *GRD*, which needs to account for *all* possible sequences that may be observed. Accordingly, the *GRD* analysis needs to consider all aspects of the recognition setting that may affect goal recognition. To facilitate this analysis, we divide a goal recognition model R into three main elements, namely the *environment,* the *actor*, and the *observer*. This division applies to most goal recognition models even though the actual representation formalism used to describe them may vary.

The *environment* describes the dynamics of the setting in which agents act, including all aspects of the model that dictate the possible behaviors of agents within it. It is common to use a compact, rather than explicit, representation of possible agent behaviors, including the set of possible goals G, the initial state I, and the set of actions A that may be executed by an agent. For deterministic settings, agent behavior is described by a plan π, which is a sequence of actions from the initial state to some goal, and $pre(\pi)$ represents a plan's prefix. For non-deterministic settings, a policy π, which is a mapping from states to actions, is used instead. A policy prefix is a policy defined only for states that are reachable from the start state [Wayllace *et al.*, 2016]. While most work on *GRD* so far uses planning domain theories [Geffner and Bonet, 2013] to represent the environment, Mirsky *et al.* 2019 use plan libraries to represent agent behavior.

Given the set of possible policies (or plans) to a goal, the *actor* component describes the set $\Pi^{leg}(g) \subseteq \Pi(g)$ of *legal policies*, the policies actors may choose to execute in order to achieve goal g. These policies are those allowed under assumptions about the behavior of actors in the system and about how they choose which action to execute at each stage.

There are various factors that affect an actor's decision on how to behave, including its familiarity with the environment (possibly reflected by its sensor model), its capabilities and preferences (*e.g.*, can the actor compute an optimal plan?), its relationship to the recognizer (*e.g.*, is it aware of its presence?), and more. For example, assuming actors are optimal, the set of legal plans are those that minimize the cost to a goal in a deterministic environment, and the set of policies that minimize the expected cost to a goal in a stochastic environment.

The *observer* models the way actors and their actions are perceived. The observer component can represent a passive recognition system or another agent that may be active in the system, aiming to interpret the actors' behavior. Either way, the *GRD* analysis focuses on the goal recognition task, and the observability as considered here describes how the activity of an actor is perceived by the observer, independently of how the actors perceive the environment.

The observer's *sensor model* S maps an execution sequence to an observation sequence $\vec{o}$ that may be emitted when performed by an actor. The simplest sensor model corresponds to the fully observable setting, where the observer can sense actions and state transitions. More generally, a sensor model may reflect partial and noisy sensing where an execution may emit different observation sequences and probabilities that may be associated with each observation. The set $\vec{O}$ of possible observation sequences and their associated probabilities are therefore induced by the sensor model S and the set of legal plans $\Pi^{leg}(G)$.

Given a goal recognition model and its sensor model, each sequence of actions is associated with (mapped to) a set of goals *satisfied* by its emitted observations. Typically, a goal can be recognized when the behavior of an actor becomes *distinctive*, i.e., it can be associated with a single goal.

2.2 Design Model

The available ways to modify a goal recognition model and the different constraints that may be imposed on the design process vary among applications and settings. For example, disallowing actions correspond to removing actions from the action set. Other modifications were proposed in the literature, including *sensor refinement* [Keren *et al.*, 2019; Wayllace *et al.*, 2020], applied in partially observable settings, which is expressed as a change in the recognition system's sensor model.

We present next a generic model to support arbitrary modifications. Given a set of goal recognition models $\mathcal{R}$, the design model describes the modifications that can be applied to a goal recognition model $R \in \mathcal{R}$ and the design objective.

Definition 1 *A design model $D = \langle \mathcal{M}, \delta, \phi, \mathcal{U} \rangle$ is a quadruple where:*

- *$\mathcal{M}$ is a finite set of atomic modifications a system can apply. A modification sequence is an ordered set of modifications $\vec{m} = \langle m_1, \ldots, m_n \rangle$ s.t. $m_i \in \mathcal{M}$ and $\vec{\mathcal{M}}$ is the set of all such sequences.*
- *$\delta : \mathcal{M} \times \mathcal{R} \to \mathcal{R}$ is a deterministic modification transition function, specifying the goal recognition model that results from applying a modification to a goal recognition model.*
- *$\phi : \vec{\mathcal{M}} \times \mathcal{R} \to \{0, 1\}$ is a constraint indicator that specifies the modification sequences that can be applied to a goal recognition model, and*
- *$\mathcal{U} : \mathcal{R} \to \mathbb{R}$ is a score used to assess a goal recognition model.*

Each modification $m \in \mathcal{M}$ can be associated with a design cost $C_D(m)$, and the cost of a sequence is the aggregated cost of its components ($\mathcal{C}_D(\vec{m}) = \sum_{i=1}^{n} C_D(m_i)$).

The constraint indicator ϕ imposes the set of modifications that are applicable in a goal recognition model R. We represent this set as $app_\phi(R) = \{m \in \mathcal{M} \mid \phi(m, R) = 1\}$. A constraint function may, for example, impose a design budget, limiting the cost (or number) of allowed modifications. With the objective of maintaining usability, one can also bound the increase in optimal costs to the goals in the modified setting. Specifically, one may require the redesign process to leave the optimal cost for plans to all goals unchanged.

The final component of the design model is the measure $\mathcal{U}$ used to assess how well goal recognition can be performed in a given model. This measure can vary between applications and settings. In our running example, we described the *worst case distinctiveness* (*WCD*) measure, first presented by Keren *et al.* (2014), which represents the maximal cost of a *non-distinctive* path, which is a path an agent can follow without revealing its goal. Additional measures were presented in the literature and will be discussed below.

2.3 *GRD* Problem Definition

Given a goal recognition model and a design model, a *GRD* model is defined as follows:

Definition 2 *A goal recognition design (*GRD*) model is given by the pair* $T = \langle R_0, D \rangle$ *where*

- R_0 *is an initial goal recognition model, and*
- D *is the* design model

The design model D imposes a set $\mathcal{R}^T \subseteq \mathcal{R}$ of goal recognition models reachable from the initial model R_0 by applying a valid modification sequence. Using a *GRD* model, the *GRD* problem aims at identifying a sequence of modifications that maximize the design objective while complying with the design constraints.

3 Environment Dynamics Overview

The initial *GRD* model [Keren *et al.*, 2014] is based on the assumption that action outcomes are deterministic. Therefore, when an actor transitions between states, it is assumed that it was the actor's intention for this transition to occur. In many realistic settings, however, this assumption does not hold, and the recognizer needs to account for the environment dynamics to be able to infer the actor's intentions.

In the running example, when an optimal actor is observed moving to the right of the initial state and assuming a deterministic environment, the actor's goal can be inferred to be the goal on the right. Blocking the ability to move straight ahead from the initial state is sufficient to minimize *WCD*. However, if the environment is stochastic such that the actor ends up in its intended cell only with 80% probability, with the remaining 20% randomly distributed among adjacent cells, a move to the right no longer guarantees that the actor's goal is revealed.

To account for such settings, Wayllace *et al.* (2017) extended *GRD* to support stochastic agent actions. Instead of minimizing the maximal non-distinctive path, *Stochastic GRD* (S-GRD) accounts for probabilistic behaviors by computing the maximal expected cost an actor may incur before its goal is revealed. Accordingly, *WCD* of an S-GRD model represents the largest expected cost incurred by the agent over all non-distinctive policy prefixes. In addition, since the *WCD* definition makes the implicit assumption that all goals have equal prior likelihood, Wayllace *et al.* (2017) formulate the *expected-case distinctiveness (ECD)* measure that weighs the possible goals based on their prior likelihood of being the true goal.

In terms of design, environment modifications that have been examined so far include disallowing actions [Keren *et al.*, 2014; Wayllace *et al.*, 2017], as demonstrated in our example. Keren *et al.* (2018; 2019) support *action conditioning*, a more general design option that can be used to induce a partial order among actions. Similarly, Mirsky *et al.* (2019) remove rules from the grammar that represents the actor's planning domain, effectively changing the preconditions and effects of actions in the model.

4 Actor Types Overview

Our discussion thus far assumed that both actor and observer have full observability and full knowledge about the environment. This section describes work that considered cases where the actor is only partially informed and may be sub-optimal. Considering cases where the observer may be partially informed is discussed in the next section.

The first account of sub-optimal actors was by Keren *et al.* (2015), where actors are *bounded sub-optimal*, with a bound by which they can divert from optimal behavior. According to this model, actors behave sub-optimally either naïvely, by following any plan within the specified bound, or intentionally, by following prefixes of optimal plans to other goals, thus obfuscating their true goal as far as possible within the cost bound. In our example, actors aiming at G_0 can conceal their objective with a diversion bound of 4 actions by following an optimal path to G_1 (moving right two steps and then up to G_1), revealing their true goal only after reaching G_1. Consequently, the design solution suggested in Figure 1 cannot guarantee *WCD* minimization. The optimal design solution for this case would require creating a longer barrier.

Keren *et al.* (2016b) show that the same *WCD* measure and the tools used for its computation can become handy to privacy-preserving actors to understand how to behave in order to obfuscate their goals. Other lines of work considered other forms of adversarial or privacy-preserving actors that try to conceal their objective. Ang *et al.* (2017) support a game-theoretic version of *GRD*, in which a recognition system can alter the environment to facilitate the early detection of attacks by strategic adversarial agents that obfuscate their targets. An adversarial *GRD* setting was also suggested by Bisson *et al.* (2011) where it is possible to dynamically set the value of environment features in order to provoke an opponent to behave in a way that reveals her intention. Masters and Sardina (2017; 2019) consider adversarial agents in continuous path planning domains. While their work does not discuss design per-se, it offers an offline analysis of goal recognition settings by creating a *probabilistic heatmap* that includes the probabilities of each goal at every step in the actor's grid. This heatmap finds the Radius of Maximum Probability (RMP), the radius within which a goal is guaranteed to be the most probable. This framework allows the actor to find a path such that the probability of an observer identifying its final destination before it has been reached, is minimised.

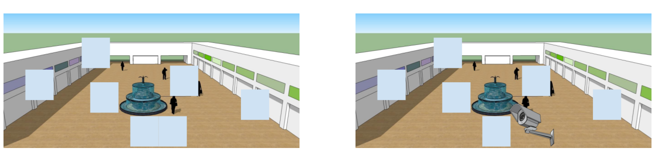
Recently, Keren *et al.* 2020 suggest a *GRD* setting where actors, which are agnostic to the recognition process, are modeled as partially informed planning agents. The recognizer has perfect information and can selectively perform *information shaping* modifications that reveal information to the actors in order to alter their behavior and induce behaviors that facilitate recognition by minimizing *WCD*.

	Environment Dynamics	Actor Type	Observer's Sensor Model	Metric	Design
Offline					
Keren *et al.* 2014	deterministic	optimal	full observability	*WCD*	action removal
Keren *et al.* 2015	deterministic	bounded suboptimal or adversarial	full observability	*WCD*	action removal
Son *et al.* 2016	deterministic	optimal	full observability	*WCD*	action removal
Keren *et al.* 2016a	deterministic	bounded suboptimal or adversarial	partial observability	*WCD*	action removal action sensor refinement
Keren *et al.* 2016b	deterministic	privacy preserving	noisy sensing	*WCD*	state sensor refinement
Ang *et al.* 2017	deterministic	adverserial	full observability	min-max value	action removal
Wayllace *et al.* 2017	stochastic actions	optimal	full observability	*WCD* ECD	action removal
Mirsky *et al.* 2019	deterministic	optimal	full observability	*WCD*	rule removal
Masters and Sardina 2019	continuous	adversarial	full observability	RMP	none
Wayllace *et al.* 2020	stochastic actions	optimal	partial observability	*WCD*	action removal state sensor refinement
Keren *et al.* 2020	deterministic	partially informed	full observability	*WCD*	information shaping
Online					
Bisson *et al.* 2011	deterministic	adversarial	full observability	goal probability	environment change
Mirsky *et al.* 2018	deterministic	optimal	full observability	goal probability	actor queries
Shvo and McIlraith 2020	deterministic	optimal	partial observability	landmarks	world-altering actions sensing actions

Table 1: summary of *GRD* models

5 Observer Models Overview

Initial works on *GRD* assume that actors are fully observable to the goal recognition system. However, many real-world applications must account for various forms of partial observability. In particular, goal recognition systems may suffer from reduced and noisy observability due to a lack of suitable sensors, insufficient sensor coverage, faulty sensors, inaccurate measurements, *etc*. Whereas in the fully observable setting, goal recognition is hampered only if an actor's behavior could fit more than one goal, when observability is partial an actor's goal can remain unrecognized even if its behavior is goal-specific.



(a) (b)

Figure 2: Observer's partial observability

Example 2 *The setting depicted in Figure 2 differs from the one described in Example 1 by accounting for partial sensor coverage (a move action ending in a blank cell is non-observable). As can be seen, the modification applied to the fully observable setting does not offer the same benefit in the partially observable setting. Even with the obstacle placed in front of the entry point, optimal agents may advance one step before their goal is revealed (Figure 2a) (*$\text{WCD} = 1$*). In Figure 2b, a sensor is placed to the right of the entry point, thus guaranteeing recognition at the first step (to the left or right), setting* WCD *to* 0.

To support settings with partially informed observers Keren *et al.* (2016a) extend *GRD* to account for goal recognition systems with partial sensor coverage. In this setting, actions can be either observable or non-observable, and sensor placement is proposed to improve recognition. Keren *et al.* (2016b) generalize the sensor model to account for non-deterministic and noisy sensor models. Sensor refinement, the act of improving the system's sensor resolution, is added to the set of possible modifications to the model. Wayllace *et al.* (2020) offer a model that accounts for partial observability and sensor refinement in stochastic domains.

Sensor refinement can also be applied to continuous domains where, as in discrete domains, the recognition system may fail because the same observation sequence may by emitted by multiple plan prefixes to different goals due to low sensor resolution [Vered and Kaminka, 2017]. In such settings, *GRD* can be applied to increase the resolution of specific parts of the model.

All the frameworks mentioned so far assume the observer is passive, forced to wait until an actor produces observations that can disambiguate its goal. Supporting a proactive observer, Bisson *et al.* (2011) offer a framework in which the observer can dynamically provoke the actor to behave in a way that reveals its intentions by setting a value of an environment variable. Knowing how it would react to the provocation, deciding when and which variable to set is modeled as a planning problem, where the value of each intervention is based on the expected reduction of the uncertainty on the actor's intentions. Similarly, Mirsky *et al.* (2018) support settings in which the observer can iteratively query an actor about some environment variable to ensure early recognition. The decision of which query to pose is based on the likelihood of the different goals that are related to the query and its potential information gain given the current probabilities of each goal. Shvo and McIlraith (2020) support an observer that can dynamically decide to sense specific environment variables or act in the environment in a way that expedites the recognition of an actor's goal. The decision of which intervention to perform is based on the analysis of each goal's *landmarks*, facts that hold for all plans that achieve that goal. At each iteration, the approach finds the most unique landmark, generates a plan to sense it, and eliminates all goals that are not associated with the sensed value.

6 Approaches to Design

We now explore the different methods that have been suggested to solve *GRD* problems. Such solutions typically need to answer two main questions: how to evaluate a given goal recognition model and how to find the best design solution.

A baseline approach to evaluate the ability to perform goal recognition involves exploring possible actor plans to each goal, and finding the maximal overlap (or expected overlap in stochastic domains) between behaviors associated to different goals using some goal recognition method (e.g. [Ramirez and Geffner, 2010]). Such an approach is only effective when possible plans can be enumerated and calculated efficiently offline. An alternative approach defers the analysis of actors' behavior until they are active in the system, deciding online about the need to intervene based on the current most probable goal [Kabanza *et al.*, 2010; Mirsky *et al.*, 2018; Shvo and McIlraith, 2020]. For cases where an actor's possible plans to each goal are bounded by the actor's maximal cost to goal, Keren *et al.* (2019) suggest a variety of compilations to classical planning that allow the offline computation of *WCD* for a variety of *GRD* settings, using a single call to an off-the-shelf classical planner.

As far as finding the best design solution, a naïve approach is an exhaustive exploration of the design options. However, the design search space may be large and more efficient design solutions were proposed in the literature.

Keren *et al.* (2019) formulate the design process as a state space search in the space of modification sequences $\vec{\mathcal{M}}$. The root node is the initial goal recognition model R_0 (and empty modification set), and the operators (edges) are the modifications $m \in \mathcal{M}$ that transition between models. Each node (modification sequence) is evaluated according to the utility measure for the problem at hand (e.g. using the *WCD* value of its corresponding model). This formulation inspired the design of different pruning approaches. Keren *et al.* (2019) suggest the *pruned-reduce* approach, which focuses the search on the current plans that maximize *WCD*, and specify conditions under which it is guaranteed to yield optimal solutions. A variation of the *pruned-reduce* approach was used in other lines of work [Mirsky *et al.*, 2019; Wayllace *et al.*, 2017]. Harman and Simoens (2019) propose the use of *action graphs*, and-or graphs that capture the relationship between actions, to determine which actions should be replaced or removed from the model in order to facilitate recognition. Son *et al.* (2016) formulate a *GRD* problem as an Answer Set Programming (ASP) and use highly optimized and effective ASP solvers to compute a solution to a *GRD* problem. To date, this solution outperforms planning-based approaches for fully observable models with optimal agents, but has not been applied to more general settings. Ang *et al.* (2017) encode the design process as a mixed-integer program (MIP) that is used to find the best observer strategy for applying modifications to the environment.

As a concluding remark, we note that *GRD* can be viewed as a form of *environment design* [Zhang *et al.*, 2009], which supports an interested party in finding optimal modifications to apply to an environment in order to maximize some utility measure. *GRD* can also be seen as a form of mechanism design, where the objective is to influence future interactions between an agent and the goal recognition system. Specifically, reducing *WCD* by eliminating actions can be seen as an aspect of social law [Shoham and Tennenholtz, 1995], with eliminated actions viewed as being made illegal.

7 Concluding Discussion

Since the early days of AI, a variety of goal recognition tools have been developed to provide automated ways to efficiently analyze agent behavior, contributing to this active and evolving field of research. Goal recognition design (*GRD*), the focus of this survey, is a novel approach that suggests a principled way to redesign a goal recognition setting in order to facilitate online goal recognition.

Although *GRD* has made substantial progress since it was first introduced, there are many challenges that have not yet been addressed. While there exists *GRD* work that supports either offline or online design solutions, there is no work that combines the two approaches. Considering the trade-off between the offline and online design options is an interesting avenue for future research. Another important issue involves incorporating into *GRD* tools more general accounts of uncertainty of goal recognition systems, especially those concerning partial information actors may have about their surroundings and about other actors in a stochastic environment with noisy sensing. Finally, the relationship between the actors and the observer, that can be characterized as collaborative, agnostic, or adversarial, deserves a more thorough investigation in the context of *GRD*.

The theoretical progress in analyzing *GRD* under various settings opens the way to exploring *GRD* in various real-world applications where goal recognition is essential and for which the design can be controlled. Specifically, *GRD* fits well with the emerging need for efficient and effective communication between humans and robots, may it be at home, at the office, or on the move. Such transfer from theory to practice requires looking into engineering solutions that can scale to environments with many behavior and design alternatives.

Acknowledgements

The research leading to these results is partially supported by Accenture LTD and by the Center for Research on Computation and Society at Harvard University.

References

[Albrecht and Stone, 2018] S.V. Albrecht and P. Stone. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence*, 258:66–95, 2018.

[Ang *et al.*, 2017] S. Ang, H. Chan, A.X. Jiang, and W. Yeoh. Game-theoretic goal recognition models with applications to security domains. In *GameSec*, 2017.

[Bisson *et al.*, 2011] F. Bisson, F. Kabanza, A.R. Benaskeur, and H. Irandoust. Provoking opponents to facilitate the recognition of their intentions. In *AAAI*, 2011.

[Carberry, 2001] S. Carberry. Technique for plan recognition. *User Modeling and User-Adapted Interaction*, 2001.

[Chakraborti *et al.*, 2017] T. Chakraborti, S. Sreedharan, Y. Zhang, and S. Kambhampati. Plan explanations as model reconciliation: Moving beyond explanation as soliloquy. In *IJCAI*, 2017.

[Chakraborti *et al.*, 2019] T. Chakraborti, A. Kulkarni, S. Sreedharan, D.E. Smith, and S. Kambhampati. Explicability? legibility? predictability? transparency? privacy? security? the emerging landscape of interpretable agent behavior. In *ICAPS*, 2019.

[Freedman and Zilberstein, 2017] R.G. Freedman and S. Zilberstein. Integration of planning with recognition for responsive interaction using classical planners. In *AAAI*, 2017.

[Geffner and Bonet, 2013] H. Geffner and B. Bonet. A concise introduction to models and methods for automated planning. *Synthesis Lectures on AI and ML, Morgan & Claypool Publishers*, 2013.

[Ha *et al.*, 2011] E. Ha, J. P. Rowe, B. W. Mott, and J.C. Lester. Goal recognition with Markov logic networks for player-adaptive games. In *AIIDE*, 2011.

[Harman and Simoens, 2019] H. Harman and P. Simoens. Action graphs for performing goal recognition design on human-inhabited environments. *Sensors*, 19(12):2741, 2019.

[Jarvis *et al.*, 2004] P. A. Jarvis, T. F. Lunt, and K. L. Myers. Identifying terrorist activity with AI plan recognition technology. In *IAAI*, 2004.

[Kabanza *et al.*, 2010] F. Kabanza, P. Bellefeuille, F. Bisson, A.R. Benaskeur, and H. Irandoust. Opponent behaviour recognition for real-time strategy games. In *PAIR*, 2010.

[Kautz, 1987] H. A. Kautz. *A Formal Theory of Plan Recognition*. PhD thesis, Bell Laboratories, 1987.

[Keren *et al.*, 2014] S. Keren, A. Gal, and E. Karpas. Goal recognition design. In *ICAPS*, 2014.

[Keren *et al.*, 2015] S. Keren, A. Gal, and E. Karpas. Goal recognition design for non-optimal agents. In *AAAI*, 2015.

[Keren *et al.*, 2016a] S. Keren, A. Gal, and E. Karpas. Goal recognition design with non-observable actions. In *AAAI*, 2016.

[Keren *et al.*, 2016b] S. Keren, A. Gal, and E. Karpas. Privacy preserving plans in partially observable environments. In *IJCAI*, 2016.

[Keren *et al.*, 2018] S. Keren, A. Gal, and E. Karpas. Strong stubborn sets for efficient goal recognition design. In *ICAPS*, 2018.

[Keren *et al.*, 2019] S. Keren, A. Gal, and E. Karpas. Goal recognition design in deterministic environments. *Journal of Artificial Intelligence Research*, 65:209–269, 2019.

[Keren *et al.*, 2020] S. Keren, H. Xu, K. Kwapong, D. C. Parkes, and B. Grosz. Information shaping for enhanced goal recognition of partially-informed agents. In *AAAI*, 2020.

[Kulkarni *et al.*, 2019a] A. Kulkarni, S. Sreedharan, S. Keren, T. Chakraborti, D. E. Smith, and S. Kambhampati. Design for interpretability. In *XAIP-ICAPS*, 2019.

[Kulkarni *et al.*, 2019b] A. Kulkarni, S. Srivastava, and S. Kambhampati. A unified framework for planning in adversarial and cooperative environments. In *AAAI*, 2019.

[Levine and Williams, 2018] S. J. Levine and B. C. Williams. Watching and acting together: concurrent plan recognition and adaptation for human-robot teams. *Journal of Artificial Intelligence Research*, 63:281–359, 2018.

[MacNally *et al.*, 2018] A. MacNally, N. Lipovetzky, M. Ramirez, and A. Pearce. Action selection for transparent planning. In *AAMAS*, 2018.

[Masters and Sardina, 2017] P. Masters and S. Sardina. Cost-based goal recognition for path-planning. In *AAMAS*, 2017.

[Masters and Sardina, 2019] P. Masters and S. Sardina. Cost-based goal recognition in navigational domains. *Journal of Artificial Intelligence Research*, 64:197–242, 2019.

[Mirsky *et al.*, 2018] R. Mirsky, R. Stern, K. Gal, and M. Kalech. Sequential plan recognition: An iterative approach to disambiguating between hypotheses. *Artificial Intelligence*, 260:51–73, 2018.

[Mirsky *et al.*, 2019] R. Mirsky, K. Gal, R. Stern, and M. Kalech. Goal and plan recognition design for plan libraries. *ACM Transactions on Intelligent Systems and Technology*, 2019.

[Pereira *et al.*, 2017] R. F. Pereira, N. Oren, and F. Meneguzzi. Landmark-based heuristics for goal recognition. In *AAAI*, 2017.

[Ramirez and Geffner, 2010] M. Ramirez and H. Geffner. Probabilistic plan recognition using off-the-shelf classical planners. In *AAAI*, 2010.

[Shoham and Tennenholtz, 1995] Y. Shoham and M. Tennenholtz. On social laws for artificial agent societies: Off-line design. *Artificial Intelligence*, 1995.

[Shvo and McIlraith, 2020] M. Shvo and S. A. McIlraith. Active Goal Recognition. In *AAAI*, 2020.

[Son *et al.*, 2016] T.C. Son, O. Sabuncu, C. Schulz-Hanke, T. Schaub, and W. Yeoh. Solving goal recognition design using ASP. In *AAAI*, 2016.

[Sukthankar *et al.*, 2014] G. Sukthankar, C. Geib, H. H. Bui, D. Pynadath, and R. P. Goldman. *Plan, activity, and intent recognition: Theory and practice*. Newnes, 2014.

[Vered and Kaminka, 2017] M. Vered and G. A. Kaminka. Heuristic online goal recognition in continuous domains. In *IJCAI*, 2017.

[Wayllace *et al.*, 2016] C. Wayllace, P. Hou, W. Yeoh, and T. C. Son. Goal recognition design with stochastic agent action outcomes. In *IJCAI*, 2016.

[Wayllace *et al.*, 2017] C. Wayllace, P. Hou, and W. Yeoh. New metrics and algorithms for stochastic goal recognition design problems. In *IJCAI*, 2017.

[Wayllace *et al.*, 2020] C. Wayllace, S. Keren, W. Yeoh, A. Gal, and E. Karpas. Accounting for partial observability in stochastic goal recognition design: Messing with the marauder's map. In *ECAI*, 2020.

[Zhang *et al.*, 2009] H. Zhang, Y. Chen, and D.C. Parkes. A general approach to environment design with one agent. In *IJCAI*, 2009.