Exploring Options for Efficiently Evaluating the Playability of Computer Game Agents

Todd Wareham and Scott Watson
Department of Computer Science
Memorial University of Newfoundland
St. John's, NL Canada A1B 3X5
Email: harold@mun.ca, saw104@mun.ca

***Abstract*—Procedural Content Generation (PCG) is an important challenge in computer game design. The practical deployment of PCG requires efficient methods for automatically evaluating the playability of generated game content. Research to date has focused on the generation and evaluation of levels within games. However, the increasing importance of automated narrative generation involving complex non-player computer game agents (for example, agents capable of exchanging items and facts with each other and human players) suggests methods for automatically evaluating the playability of groups of such agents should also be investigated. In this paper, we propose an augmented finite-state machine model for agents with item and fact exchange capabilities and use computational complexity analysis to explore under what restrictions efficient playability evaluation of groups of such agents is and is not possible.**

I. INTRODUCTION

Given the time and cost involved with the human design of computer games, the ability to automatically generate game content (especially content whose level of difficulty can be easily adjusted to the abilities of human players) is an important problem in computer game design. Research on this problem currently goes under the name of procedural content generation (PCG) [1], [2]. By definition, PCG concerns itself with the generation of all aspects of a game (*e.g.*, game level structure) except computer-controlled non-player character (NPC) agents and the game engine itself [1, p. 172].

An important subproblem of PCG is automatically assessing the human playability of generated game content with respect to aspects ranging from basic "hard" constraints (*e.g.*, a player can actually finish a level or achieve the goal or goals associated with a level) to more psychologically-based "soft" constraints (*e.g.*, gameplay on a level maintains a rhythmic variation in difficulty) [1], [2]. As the problem of determining whether given levels can be completed is NP-hard (and hence is not efficiently solvable for all possible inputs) for many classical games [3]–[6], efficient assessments of playability done within current implementations of PCG must of necessity be driven by restricted exhaustive, evolutionary, or heuristic-based simulation searches of the gamespace which are not guaranteed to always correctly assess playability but are fast and judge correctly most of the time [1].

As mentioned above, PCG does not consider issues related to the automatic generation or evaluation of NPC game agents; this is not surprising, given the widespread opinion that current techniques based on finite-state machines seem to work quite well in released computer games and thus the problem of generating such agents has been effectively solved [7], [8]. However, though such classical models of game agents are sufficient for modeling short-term action-based interactions with human players, they are not satisfactory for modeling more socially realistic agent-player interactions that take place over longer (possibly disjoint) periods of time, require a larger degree of memory on the part of agents, and involve the maintenance of collections of items and facts by agents which can be both exchanged with and used in defining behavior with respect to other agents and human players.

Given that such socially complex agents will probably be required as part of current efforts towards automatically integrating and maintaining narratives within games [9], [10], PCG should be extended to encompass the generation of groups of such agents. A first step in this would be to automatically assess the playability of a given group of agents. While it seems likely that variations of the search techniques described above could be adapted for this purpose, it would be good to know if efficient correctness-guaranteed methods are available and, if so, in what circumstances.

Using techniques from classical complexity theory [11], we show that evaluating the playability of a group of computer game agents (in particular, determining if a human player can interact with a group of agents to obtain a specified goal-set of items and facts) is NP-hard even in the case where there is only one given agent. This holds true regardless of whether or not there is a time limit on achieving the goal. These results indicate that automatically evaluating the playability of computer game agents is computationally intractable unless very special conditions apply. This raises the question of which conditions characterize those situations in which agent playability evaluation is tractable. An answer to this question can inform computer game designers about the conditions that allow PCG relative to game agents, whether by exhaustive or heuristic search- or simulation-based methods, to be feasible, as well as give a more robust understanding of the situations in which such approaches do and do not work.

In order to find conditions for tractability, we performed a parameterized complexity analysis [12] of the problem of evaluating agent playability. Our analysis reveals that only certain restrictions on both the agents and human-agent interactions render playability evaluation tractable. Though these results are derived for a specific model of game agents and playability, we show that they also apply to a broad class of models.

The remainder of this paper is organized as follows. In Section II, we present an augmented finite-state machine model of game agents that can exchange items and facts with other agents and human players and formalize playability evaluation for such agents. Section III demonstrates the intractability of this problem. Section IV describes a methodology for identifying conditions for tractability, which is then applied in Section V to identify such conditions for agent playability evaluation. In order to focus on the implications of our results for computer game design (as well as deal with the space limitations of a conference paper), all proofs of results are given in an online supplement.[1] Finally, our conclusions and directions for future work are given in Section VI.

II. Formalizing Agent Playability Evaluation

At a minimum, an agent capable of exchanging items and facts with another agent (which could be a human player) should be able to do the following:

- Maintain an internal state as well as collections of personal items and facts;
- Perform actions (and possibly change internal state) in response to another agent's actions and offered items and facts; and
- As part of a performed action, give in return some of its own personal items and facts to that other agent.

Following [13], we distinguish items and facts as follows: there can be at most one copy of an item in a game at any time (*i.e.*, an item can be possessed by at most one agent or human player) but there can be any number of copies of a fact (*i.e.*, any number of agents or human players can possess the same fact). The sets of facts and items possessed by an agent effectively function as a finite but dynamic memory of past encounters.

There are many ways of modeling agents with the requisite abilities described above. In this paper, we will augment the finite-state machine model typically used in implementing game agents [14]. Recall that for any set S, 2^S denotes the set of all possible subsets of S (including the empty set $\emptyset$). Define an **augmented finite machine (AFSM)** (see Figure 1) relative to game-overall action-, item-, and fact-sets A^G, I^G, and F^G as a 2-tuple $\langle Q, \delta \rangle$ where Q is a set of states and $\delta \subseteq Q \times A^G \times 2^{I^G} \times 2^{F^G} \times A^G \times 2^{I^G} \times 2^{F^G} \times Q$ is a state-transition relation. A transition $(q, a, I, F, a', I', F', q') \in \delta$ of an AFSM M can be interpreted as an interaction between M and another agent in which that other agent performs action a with item- and fact-sets I and F offered to M and M responds in turn via action a' with (1) a change from state q to state q' and (2) item- and fact-sets I' and F' being given to the other agent. Any unspecified proposed action and offered item- and fact-sets relative to a state q whose result is not explicitly stated in δ is assumed to loop back on q with no effect, *e.g.*, M ignores the offered amulet and mumbles under its breath. There are many possible ways of specifying determinism and non-determinism relative to AFSM; in this paper, we will focus on AFSM that are **offered (o-) deterministic**, *i.e.*, for any given q, a, I, and F, there is at most one $(q, a, I, F, a', I', F', q') \in \delta$.

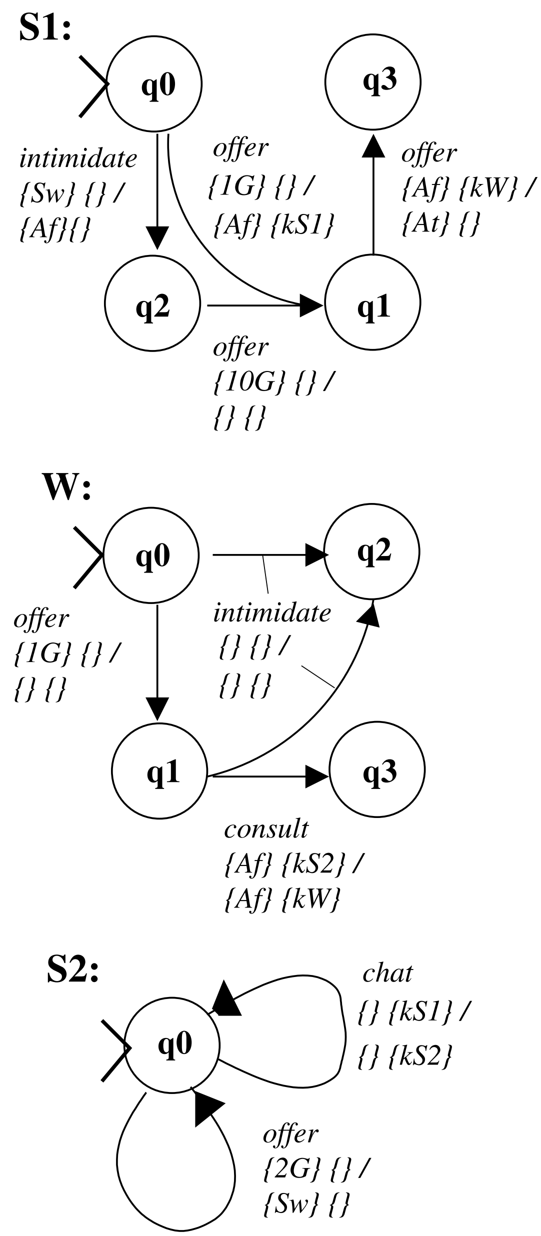

Fig. 1. Three Example Augmented Finite-State Machine (AFSM) agents

Examples of three possible AFSM representing two shopkeepers $S1$ and $S2$ and a wizard W are shown in Figure 1. These AFSM are defined relative to the action-, item-, and fact-sets {*chat, consult, intimidate, offer*}, {*false amulet (Af), true amulet (At), gold piece (G), sword (Sw)*}, and {*know shopkeeper #1 (kS1), know shopkeeper #2 (kS2), know wizard (kW)*}, respectively. Each transition $(q, a, I, F, a', I', F', q')$

[1] http://www.cs.mun.ca/∼harold/Papers/CIG14supp.pdf

Interaction-sequence #1

Interaction	P	$S1$	$S2$	W
–	$\{2G\},\{\}$	$q0:\{Af,At\},\{kS1\}$	$q0:\{Sw\},\{kS2\}$	$q0:\{\},\{kW\}$
$S1$: *offer* $\{1G\},\{\}$	$\{1G,Af\},\{kS1\}$	$q1:\{1G,At\},\{kS1\}$	”	”
$S2$: *chat* $\{\},\{kS1\}$	$\{1G,Af\},\{kS1,kS2\}$	”	”	”
W: *offer* $\{1G\},\{\}$	$\{Af\},\{kS1,kS2\}$	”	”	$q1:\{1G\},\{kW\}$
W: *cnslt* $\{Af\},\{kS2\}$	$\{Af\},\{kS1,kS2,kW\}$	”	”	$q3:\{1G\},\{kW\}$
$S1$: *offer* $\{Af\},\{kW\}$	$\{At\},\{kS1,kS2,kW\}$	$q3:\{1G,Af\},\{kS1\}$	”	”

Interaction-sequence #2

Interaction	P	$S1$	$S2$	W
–	$\{20G\},\{\}$	$q0:\{Af,At\},\{kS1\}$	$q0:\{Sw\},\{kS2\}$	$q0:\{\},\{kW\}$
$S2$: *offer* $\{2G\},\{\}$	$\{18G,Sw\},\{\}$	”	$q0:\{2G\},\{kS2\}$	”
$S1$: *intim* $\{Sw\},\{\}$	$\{18G,Sw,Af\},\{\}$	$q2:\{At\},\{kS1\}$	”	”
W: *offer* $\{1G\},\{\}$	$\{17G,Sw,Af\},\{\}$	”	”	$q1:\{1G\},\{kW\}$
W: *cnslt* $\{Af\},\{\}$	”	”	”	”
$S1$: *offer* $\{10G\},\{\}$	$\{7G,Sw,Af\},\{kS1\}$	$q1:\{10G,At\},\{kS1\}$	”	”
$S2$: *chat* $\{\},\{kS1\}$	$\{7G,Sw,Af\},$ $\{kS1,kS2\}$	”	”	”
W: *cnslt* $\{Af\},\{kS2\}$	$\{7G,Sw,Af\},$ $\{kS1,kS2,kW\}$	”	”	$q3:\{1G\},\{kW\}$
$S1$: *offer* $\{Af\},\{kW\}$	$\{7G,Sw,At\},$ $\{kS1,kS2,kW\}$	$q3:\{10G,Af\},\{kS1\}$	”	”

Interaction-sequence #3

Interaction	P	$S1$	$S2$	W
–	$\{2G\},\{\}$	$q0:\{Af,At\},\{kS1\}$	$q0:\{Sw\},\{kS2\}$	$q0:\{\},\{kW\}$
$S2$: *offer* $\{2G\},\{\}$	$\{Sw\},\{\}$	”	$q0:\{2G\},\{kS2\}$	”
$S1$: *intim* $\{Sw\},\{\}$	$\{Sw,Af\},\{\}$	$q2:\{At\},\{kS1\}$	”	”
W: *intim* $\{\},\{\}$	”	”	”	$q2:\{\},\{kW\}$
$S2$: *intim* $\{\},\{\}$	”	”	”	”

Fig. 2. Three Example AFSM Agent – Human Player Interaction-sequences

is written as an arrow between q and q' with the label "$a\{I\}\{F\}/\{I'\}\{F'\}$", *i.e.*, a' is ignored. For example, $S1$ has a transition between $q0$ and $q2$ such that $S1$ hands over the fake amulet when intimidated by another agent with a sword.

We define the execution of interactions of an AFSM $M = \langle Q, \delta \rangle$ with another agent X as follows. A transition $(q, a, I, F, a', I', F', q')$ is **enabled** relative to $M = \langle Q, \delta \rangle$ and X, where M and X currently possess the items and facts in sets I_M, F_M, I_X, and F_X, respectively, if:

1) $(q, a, F, I, a', F', I', q') \in \delta$;
2) M is currently in state q;
3) $I \subseteq I_X$ and $F \subseteq F_X$; and
4) $I' \subseteq I_M \cup I$ and $F' \subseteq F_M$.

The **execution** of a transition $(q, a, I, F, a', I', F', q')$ that is enabled relative M and X has the following effects:

1) The state of M is set to q';
2) I_X is set to $(I_X - I) \cup I'$;
3) F_X is set to $F_X \cup F'$; and
4) I_M is set to $(I_M \cup I') - I'$.

In this paper, for simplicity, we only consider the case in which the other agent X is a human player, *i.e.*, agent actions can only be triggered by human players. Three possible sequences of interactions of the AFSM in Figure 1 with a human player are shown in Figure 2. Note that in each such interaction-sequence, the players and agents start with specified item- and fact-sets, each agent starts in a designated state $q0$, and if a player temporarily stops interacting an agent M left in state q with current item- and fact-sets I and F, the next interaction of the player with M resumes with M in state q with current item- and fact-sets I and F.

There are many possible ways of formalizing playability of a group of AFSM relative to a human player [13]. The many aspects of playability can be envisioned as various hard (inviolable) and soft (violable) constraints. Example hard and soft constraints are, respectively, that a specified goal must be achieved and that the interactions in any goal-achieving interaction-sequence should incorporate as many of the actions allowable to agents as possible. Assessments of playability are based on the degree to which these constraints can be satisfied by a human player interacting with the given agents. For simplicity, let us focus on minimum playability with respect to hard constraints, *i.e.*, whether are not a human player is able to interact with a given set of agents to obtain specified goal-sets of facts and items.

Given the above, we can now formalize game agent playability evaluation as follows:

AFSM AGENT PLAYABILITY EVALUATION (APE)
Input: A set $A = \{a_1, \ldots, a_n\}$ of AFSM with associated initial item- and fact-sets $\{I^0_{a_1}, \ldots, I^0_{a_n}\}$ and $\{F^0_{a_1}, \ldots F^0_{a_n}\}$, initial player item- and fact-sets I^0_P and F^0_P, goal item- and fact-sets I_G and F_G, and a positive integer t.
Question: Can the player obtain I_G and F_G by engaging in at most t interactions with the agents in A?

With respect to the goal consisting of having the true amulet and knowing the wizard, the first and second interaction-sequences in Figure 2 achieve the goal within 5 and 8 interactions, respectively, while the third interaction-sequence does not achieve the goal and moreover cannot be extended by any sequence of interactions to achieve the goal.

III. AGENT PLAYABILITY EVALUATION IS INTRACTABLE

In this section, we address whether or not agent playability evaluation can be done efficiently relative to the model described in Section II. Following general practice in Computer Science [11], we define efficient solvability as being solvable in the worst case in time polynomially bounded in the input size, and show that a problem is not polynomial-time solvable by proving it to be at least as difficult as the hardest problems in problem-class NP, *i.e.*, NP-hard (see [11] for details).

Result A: APE is NP-hard.

Modulo the conjecture $P \neq NP$ which is widely believed to be true [15], the above shows that APE is not polynomial-time solvable. Note that this result holds even in the very restricted case in which the player only interacts with a single agent, *i.e.*, $|A| = 1$, and an unlimited number of interactions between the player and that agent is allowed, *i.e.*, $k = \infty$.

IV. A METHOD FOR IDENTIFYING TRACTABILITY CONDITIONS

A computational problem that is intractable for unrestricted inputs may yet be tractable for non-trivial restrictions on the input. This insight is based on the observation that some NP-hard problems can be solved by algorithms whose running time is polynomial in the overall input size and non-polynomial only in some aspects of the input called *parameters*. In other words, the main part of the input contributes to the overall complexity in a "good" way, whereas only the parameters contribute to the overall complexity in a "bad" way. In such cases, the problem Π is said to be **fixed-parameter tractable** for that respective set of parameters. The following definition states this idea more formally.

Definition 1: Let Π be a problem with parameters $k_1, k_2, \ldots$. Then Π is said to be *fixed-parameter (fp-) tractable* for parameter-set $K = \{k_1, k_2, ...\}$ if there exists at least one algorithm that solves Π for any input of size n in time $f(k_1, k_2, ...)n^c$, where $f(\cdot)$ is an arbitrary function and c is a constant. If no such algorithm exists then Π is said to be *fixed-parameter (fp-) intractable* for parameter-set K.

In other words, a problem Π is fp-tractable for a parameter-set K if all superpolynomial-time complexity inherent in solving Π can be confined to the parameters in K. In this sense the "unbounded" nature of the parameters in K can be seen as a reason for the intractability of the unconstrained version of Π.

There are many techniques for designing fp-tractable algorithms [12], [16], and fp-intractability is established in a manner analogous to classical polynomial-time intractability by proving a parameterized problem is at least as difficult as the hardest problems in one of the problem-classes in the W-hierarchy $\{W[1], W[2], ...\}$ (see [12] for details). Additional results are typically implied by any given result courtesy of the following lemmas:

Lemma 1: [17, Lemma 2.1.30] If problem Π is fp-tractable relative to parameter-set K then Π is fp-tractable for any parameter-set K' such that $K \subset K'$.

Lemma 2: [17, Lemma 2.1.31] If problem Π is fp-intractable relative to parameter-set K then Π is fp-intractable for any parameter-set K' such that $K' \subset K$.

Observe that it follows from the definition of fp-tractability that if an intractable problem Π is fp-tractable for parameter-set K, then Π can be efficiently solved even for large inputs, provided only that the values of all parameters in K are relatively small. This strategy has been successfully applied to a wide variety of intractable problems (see [12], [18] and references). In the next section we report on our investigation of whether or not the same strategy may be used to render the problem APE tractable.

V. WHAT MAKES AGENT PLAYABILITY EVALUATION TRACTABLE?

The AFSM agent playability evaluation problem has several parameters whose restriction could conceivably render agent playability evaluation tractable. An overview of the parameters that we considered in our fp-tractability analysis is given in Table I. These parameters can be divided into three groups:

1) Restrictions on the game agents;
2) Restrictions on the human player; and
3) Restrictions on the game itself.

In the remainder of this section, we will assess the fp-tractability of APE relative to all parameters in Table I (Section V-A), note how these results apply in more general settings (Section V-B), and discuss the implications of these results for computer game design (Section V-C).

TABLE I
FIXED-PARAMETER INTRACTABILITY RESULTS FOR THE AGENT PLAYABILITY EVALUATION PROBLEM

Parameter		Result				
Description	Name	B	C	D	E	F
AGENTS:						
# agents	$\|A\|$	–	P	P	P	1
max # items / agent	i_A	0	1	1	1	–
max # facts / agent	f_A	3	–	–	–	–
max # items / interaction	i_I	0	1	1	2	1
max # facts / interaction	f_I	P	–	2	2	2
max # states / agent	$\|Q\|$	2	–	–	–	P
max # interactions / state	$\|I\|$	1	–	2	–	–
PLAYER:						
max # items / player	i_P	0	–	–	–	–
max # facts / player	f_P	P	–	–	P	P
GAME:						
max # interactions in game	t	P	P	–	P	P
max # items in goal	i_G	0	0	0	0	0
max # facts in goal	f_G	1	1	1	1	1

A. *Results*

Our intractability results are summarized in Table I. Each column describes an intractability result that holds relative to the parameter-set consisting of all parameters whose entries in that column are not dashes ("–"); if the result holds when a non-dashed parameter has constant value c, this indicated by an entry for that parameter with the value c. Result B is notable because it, when combined with results implied by Lemma 2, establishes the intractability of APE with respect to all subsets of the considered parameters that do not include $|A|$; the intractability of many (but not all) of those remaining subsets including $|A|$ is then established by Results C–F.

At present, we have a lone tractability result:

Result G: APE is fp-tractable for $\{|A|, |I|, t\}$.

Note that Results B, D, F, and G, combined with those implied by Lemmas 1 and 2, completely characterize the parameterized complexity of APE relative to each subset of parameters in the set $\{|A|, i_I, f_I, |I|, t, i_G, f_G\}$. Note that as APE is fp-intractable for $\{|I|, t\}$ $\{|A|, |I|\}$, and $\{|A|, t\}$, by Results B, D, and F, respectively, the parameter-set in Result G is minimal in the sense that no parameter in that set can be deleted to yield fp-tractability.

B. *Generality of Results*

Our intractability results, though defined relative to an admittedly restricted conception of game agent and human-agent interaction, have a remarkable generality. Observe that the model which these results hold are in fact restricted versions of more realistic alternatives, *e.g.*,

- deterministic AFSM are special cases of both nondeterministic and probabilistic AFSM (restrict the amount non-determinism to nothing and you have determinism; restrict all actions to have probability of execution 1.0 if their triggering conditions are satisfied and you have determinism);
- player-activated AFSM are special cases of autonomous AFSM (restrict ability of AFSM to act without player interaction);
- games composed of only player / agent interaction-moves are special cases of games that require players to seek out (possibly autonomously moving) agents before they can interact with them (restrict agents to all stay in one place); and
- the simplified AFSM agent model defined in Section II is a special case of agent models that possess more complex capabilities.

Intractability results for these more game-realistic alternatives then follow from the well-known observation in computational complexity theory that intractability results for a problem Π also hold for any problem Π' that has Π as a special case and can hence solve Π (suppose Π is intractable; if Π' is tractable, then it can be used to solve Π efficiently, which contradicts the intractability of Π – hence, Π' must also be intractable).

Our fp-tractability result is more fragile, as innocuous changes to agent or game models may in fact violate assumptions critical to the operation of the algorithm underlying this result. For now, we can say that as our fp-tractability results depend only on the combinatorics of the possibilities inherent in player-agent interaction and require only that a player-agent interaction can be checked for validity and performed in time polynomial in the sizes of the entities involved in that interaction, these results apply relative to playability evaluation for *all* choices of agent and game models whose individual player-agent interactions are polynomial-type verifiable relative to these models.

C. Discussion

We have found that evaluating agent playability is NP-hard (Result A). This NP-hardness holds for a basic agent model and a minimal playability condition that a human player can attain a specified goal by interacting with the given group of agents, and even when that group consists of a single agent. This intractability result underscore the computational difficulty of evaluating agent playability by any means possible, including currently-used automated search or simulated-play-based processes (see [1], [2] and references).

To our knowledge, no explicit conjectures about the sources of computational difficulty in agent playability evaluation have been made in the literature. It seems reasonable to conjecture that restrictions on the number and/or complexity of the agents involved, the number of items or facts that can be held by an agent or player or that can be involved in any human-agent interaction, the goal of said interactions, or the maximum number of interactions that can take place should render agent playability evaluation tractable. However, no single one or indeed many possible combinations of these restrictions can yield tractability, even when the parameters involved are restricted to very small constants (Results B–F).

The one exception that we have found to date is that of simultaneously restricting $|A|$, $|I|$, and t (Result G). Though this may initially seem of limited interest in that it overly restricts the form of games whose playability can be checked efficiently, it actually suggests several reasonable ways in which games can be decomposed into sub-games whose playability can be checked efficiently. For example, a long game could be decomposed into several shorter ones (restrict t). Alternatively, the game could be structured such that only a very small number of agents or player-agent interactions are necessary and/or relevant to achieving the goal (restrict $|A|$ and/or $|I|$); this could be done while preserving a an apparently larger and varied game environment by embedding the goal-relevant set of agents and interactions within a large goal-irrelevant set of agents and interactions, *e.g.*, only a few shopkeepers, wizards, or travellers are worth talking to and only about specific matters.

To conclude this section, we would like to sketch a potentially very useful application of our results. Observe that these results demonstrate fixed-parameter tractability and intractability of playability evaluation relative to various sets of parameters for any computing model equivalent (in both power and limitations) to a Turing machine; moreover, observe that the problem of evaluating whether a game is playable cannot be computationally easier than the problem of actually playing that game (as one can use any sequence of game-interactions that are playable to verify playability). Given the hypothesis that the human brain is equivalent in power and limitations to a Turing machine (which seems reasonable (see [19]–[21] and references)), our results thus also apply to human cognitive computations underlying gameplay. When paired with the bounded computational power of human cognition, this has the following implications:

1) Restricting groups of parameters for which playability is known to be intractable will not affect the perceived human difficulty of gameplay (as there no conceivable fixed-parameter algorithms relative to those parameters which human beings could exploit).
2) Only restrictions of groups of parameters for which playability is known to be tractable might affect the perceived human difficulty of gameplay (and only for those sets of parameters whose associated fixed-parameter algorithms are both actually used or discoverable by human beings).

Hence, finding those restrictions under which playability evaluation is fixed-parameter tractable would allow game designers not only to more completely characterize (and exploit in computations that must search over) the space of human-playable games but also to better personalize gameplay (by delineating those aspects of the game whose manipulation leads to well-defined changes in the difficulty of gameplay for humans). Moreover, by explicitly characterizing the boundary between games that are too easy and those that are too hard (to design, verify, and actually play), parameter-sets for which playability evaluation is fixed-parameter tractable may underlie productive gameplay itself, in terms of both automated design and human play of games.[2]

It has already been conjectured that computationally plausible models of human cognition exploit (via fixed-parameter algorithms) restrictions in the world as well as the human cognitive system (the FP-Cognition Thesis [20]). Given the above, perhaps one can propose that productive gameplay similarly exploits restrictions in the game world as well as the human cognitive system (an FP-Gaming Thesis, if you will). The validation of this thesis will require both more theoretical elaboration and psychological experimental work; however, given the potential benefits sketched above, such effort may be most worthwhile.

VI. Conclusions

We have presented a formal characterization of the problem of game agent playability evaluation relative to a simple but promising augmentation of the classic finite-state machine model of game agents. Our complexity analysis reveal that, while this problem is computationally intractable in general, there are conditions that render it tractable. Knowledge of this and other such conditions can be exploited in computer game design to create efficient procedural content generation

[2]These consequences follow only if the playability evaluation algorithms underlying these fixed-parameter tractability results explicitly produce playable interaction-sequences as part of the evaluation process. While this need not be the case in general, it is true of the algorithm underlying Result G in this paper.

methods with respect to more complex and interesting gameplay involving player interactions with more socially realistic game agents. We also conjecture that such knowledge will help in creating games whose level of difficulty is not only more appropriate to human players but can also be efficiently and appropriately customized to the abilities of those players.

In future research, we plan to explore the effect of additional types of restrictions on the computational complexity of the model of agent playability evaluation described in this paper, as well as the effects of these and existing restrictions on models incorporating more powerful agents (*e.g.*, agents that are self- rather than player-activated; agents whose transitions are triggered by arbitrary Boolean formulas over offered items and facts) relative to both minimal and broader conceptions of playability. Given the tantalizing links between human cognitive and gaming abilities hypothesized at the end of in Section V-C, in the longer term, we also intend to investigate connections between playable computer game design and theories proposed within cognitive science for gaming-related abilities such as belief maintenance, intention recognition, and analogical problem solving, with an aim to using complexity analyses of these [19], [22]–[24] and other [21] theories from cognitive science to efficiently incorporate more psychologically realistic models of agents and aspects of playability into computer game design.

Acknowledgments

The authors would like to thank Rod Byrne, Andrew Vardy, and Wolfgang Banzhaf for encouraging them to embark on this research. TW was supported by NSERC Discovery Grant 228104 and SW was supported by a Dean's Doctoral Award and a Graduate Student Fellowship from the MUN School of Graduate Studies.

References

[1] J. Togelius, G. N. Yannakakis, K. O. Stanley, and C. Browne, "Search-based Procedural Content Generation: A Taxonomy and survey," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 3, no. 3, pp. 172–186, 2011.

[2] G. N. Yannakakis and J. Togelius, "Experience-driven Procedural Content Generation," *IEEE Transactions on Affective Computing*, vol. 2, no. 3, pp. 147–161, 2011.

[3] M. Forišek, "Computational Complexity of Two-dimensional Platform Games," in *Fun with Algorithms*, ser. Lecture Notes in Computer Science, P. Boldi, Ed., no. 6099. Berlin: Springer, 2010, pp. 214–227.

[4] G. Aloupis, E. D. Demaine, A. Guo, and G. Viglietta, "Classic Nintendo Games are (Computationally) Hard," 2012.

[5] J. Lynch, "Collecting Things Under Time Pressure is Hard," *Tiny Transactions on Computer Science*, vol. 1, 2012.

[6] G. Viglietta, "Lemmings is PSPACE-complete," *arXiv preprint arXiv:1202.6581*, 2012.

[7] A. Nareyek, "Game AI is Dead. Long Live Game AI!" *IEEE Intelligent Systems*, vol. 22, no. 1, pp. 9–11, 2007.

[8] G. N. Yannakakis, "Game AI Revisited," in *Proceedings of the 9th Conference on Computing Frontiers*. ACM, 2012, pp. 285–292.

[9] M. Riedl, D. Thue, and V. Bulitko, "Game AI as Storytelling," in *Artificial Intelligence for Computer Games*, P. A. González-Calero and M. A. Gómez-Martín, Eds. Berlin: Springer, 2011, pp. 125–150.

[10] M. O. Riedl and V. Bulitko, "Interactive Narrative: An Intelligent Systems Approach," *AI Magazine*, vol. 34, no. 1, pp. 67–77, 2013.

[11] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA: W.H. Freeman, 1979.

[12] R. Downey and M. Fellows, *Fundamentals of Parameterized Complexity*. Berlin: Springer, 2013.

[13] S. Watson, W. Banzhaf, and A. Vardy, "Automated Design for Playability in Computer Game Agents," 2014.

[14] M. Dawe, S. Gargolinski, L. Dicken, T. Humphries, and D. Mark, "Behavior Selection Algorithms: An Overview," in *Game AI Pro: Collected Wisdom of Game AI Professionals*, S. Rabin, Ed. Boca Raton, FL: CRC Press, 2014, pp. 47–60.

[15] L. Fortnow, "The Status of the P Versus NP Problem," *Communications of the ACM*, vol. 52, no. 9, pp. 78–86, 2009.

[16] R. Niedermeier, *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.

[17] T. Wareham, *Systematic Parameterized Complexity Analysis in Computational Phonology*. Ph.D. thesis, Department of Computer Science, University of Victoria, 1999.

[18] U. Stege, "The Impact of Parameterized Complexity to Interdisciplinary Problem Solving," in *The Multivariate Algorithmic Revolution and Beyond*, ser. Lecture Notes in Computer Science, H. L. Bodlaender, R. Downey, F. V. Fomin, and D. Marx, Eds. Berlin: Springer, 2012, no. 7370, pp. 56–68.

[19] I. van Rooij, *Tractable Cognition: Complexity Theory in Cognitive Psychology*. Ph.D. thesis, Department of Psychology, University of Victoria, 2003.

[20] ——, "The Tractable Cognition Thesis," *Cognitive Science*, no. 32, pp. 939–984, 2008.

[21] I. van Rooij and T. Wareham, "Parameterized Complexity in Cognitive Modeling: Foundations, Applications, and Opportunities," *Computer Journal*, vol. 51, no. 3, pp. 385–404, 2008.

[22] I. van Rooij, J. Kwisthout, M. Blokpoel, J. Szymanik, T. Wareham, and I. Toni, "Intentional Communication: Computationally Easy or Difficult?" *Frontiers in Human Neuroscience*, no. 5, 2011.

[23] M. Blokpoel, J. Kwisthout, T. P. van der Weide, T. Wareham, and I. van Rooij, "A Computational-level Explanation of the Speed of Goal Inference," *Journal of Mathematical Psychology*, vol. 57, no. 3/4, pp. 117–133, 2013.

[24] T. Wareham, P. Evans, and I. van Rooij, "What Does (and Doesn't) Make Analogical Problem Solving Easy? A Complexity-Theoretic Investigation," *Journal of Problem Solving*, vol. 3, no. 2, pp. 30–71, 2011.