Leveraging efficient planning and lightweight agent definition: a novel path towards emergent narrative

Conference Paper

Author(s):
Raymond, Henry; Knobloch, Sven; Zünd, Fabio; Sumner, Robert W.; Magnenat, Stéphane

Publication date:
2020-10

Permanent link:
https://doi.org/10.3929/ethz-b-000439084

Leveraging efficient planning and lightweight agent definition: a novel path towards emergent narrative

Henry Raymond, Sven Knobloch, Fabio Zünd, Robert W. Sumner, Stéphane Magnenat[1]
[1] Game Technology Center, Dept. Computer Science, ETH Zurich
Universitätstrasse 25, UNH F3, 8092 Zürich, Switzerland
firstname.lastname@inf.ethz.ch

Abstract

Emergent narrative has the ability to unlock the true potential of interactive media, moving beyond pre-scripted, fixed storylines. Existing implementations of emergent narrative achieve their results through complex rule systems and agent representations, which entail high authoring workload that limit the feasible scope of storyworlds. In this paper, we propose an approach that instead aims at leveraging efficient planning to achieve similar results, using Monte Carlo Tree Search and efficient data structures. This allows for abstraction and modularization of agent behavior, and endows agents with a theory of mind by letting them plan for each other. This greatly simplifies agent definition and removes the need to explicitly encode intentions. We show that competitive, collaborative and sustainable behaviors emerge in our system, without the explicit definition of such behaviors. Based on these preliminary results, we discuss necessary steps to turn our approach into an applicable emergent narrative system.

Procedural storytelling has the ability to unlock the true potential of interactive media and is especially promising for video games. Unlike the fixed storylines of books, movies, and many video games, procedural storytelling can exploit high and affordable computational power to adapt to user preferences and create unique and personal content. However, in commercially-successful video games, most instances of procedural storytelling still heavily rely on prescripted storylines and enforce key events (e.g. 80 days, Heaven's Vault[1]). This does not fully capitalize on the potential of procedural storytelling, as many aspects of the experienced story are predetermined, denying the player true agency.

Emergent narrative, contrarily, is an approach to procedural storytelling that affords true agency to the player. Following James Ryan's definition (Ryan 2018, ch. 2), it is narrative that emerges from the simulation of character activity. Crucially, it emerges without continuous guidance or manipulation of the simulation. Instead, the interplay of virtual characters with their environment, each other, and possibly human players, generates events that can form a narrative.

[1]https://www.inklestudios.com/

To date, however, few games intended for a general audience employ emergent narrative. A noteworthy example is Dwarf Fortress (Short and Adams 2019, ch. 13). Works commonly associated with emergent narrative, such as Caves of Qud (Grinblat and Bucklew 2017), Façade (Mateas and Stern 2003) and Prom Week (McCoy et al. 2012), still have some form of story guidance. This shows that achieving true emergent narrative is a difficult problem, and its limited adoption suggests that many challenges are still unsolved.

Aylett has stated that a key risk of emergent narrative is that it does not emerge (Aylett 1999). The generated events may be mathematically unique, but often lack any perceptual interestingness. The goal of emergent narrative should therefore be to achieve a high level of *tellability*, as studied by Marie-Laure Ryan (Ryan 1987).

James Ryan points out that any emergent narrative system is governed by a layer of curation, which extracts tellable stories from an impartial simulation (Ryan 2018). Ryan's point of view lets us split emergent narrative into two separate challenges: simulation and curation. It follows, that the simulation should not itself aim for tellability, but instead provide a fertile ground for curation. In our eyes that means generating a variety of non-trivial behaviors and interactions between agents. In this paper, we present our approach to simulation.

Some existing approaches, such as Façade and Prom Week, achieve complex agent behavior by simulating emotions, intentions, and relationships. They rely on complex rule systems and state representations that require costly manual authoring and tuning. For instance, Prom Week's social schema has about 5000 "social considerations", which required extensive manual balancing (Ryan 2018, p. 123). Hence, due to its workload, the authoring and tuning of agent behavior is a central limiting factor for many emergent narrative systems. Consequently, addressing its scope and ease-of-use is essential for a more broad appeal of emergent narrative in real-world applications.

Other approaches, such as Dwarf Fortress, use simpler agent behaviors. In these, the richness arises from the interaction of larger numbers of simple agents with multifaceted environmental simulations. In that sense, they behave as complex systems, similar to Conway's Game of Life

in cellular automata (Gardner 1970). We believe that creating complex systems out of simple rules is key in creating easy-to-author simulations for emergent narrative.

Human-like behavior and reasoning, especially in a social context, is often inherently defined by long-term considerations of cost and benefit. Many existing implementations of emergent narrative are thus heavily impaired by the limited timeframe of agent actions, compelling them to explicitly encode long-term processes into short-term rules and states. Our hypothesis is that deep, long-term planning can remove the need of such explicit encodings, greatly simplifying action definitions and state representations. Simpler actions with less encoded reasoning permits more flexibility in terms of modularization and abstraction, two key dimensions of the simulation layer (Ryan 2018, ch. 4.1).

Another key aspect of social behavior is the consideration and prediction of the intentions of others, an ability which psychology describes as part of the theory of mind (Premack and Woodruff 1978). Existing implementations infer intentions of others through belief-based reasoning or make agents' intentions perceivable to others. These approaches need extensive ad-hoc modelling to emulate a theory of mind. Contrarily, in this paper we show that a theory of mind naturally arises when each agent considers the long-term planning of other relevant agents.

Overall, in this paper we propose a novel simulation design for emergent narrative that overcomes many limitations of existing approaches. This design relies on simple action definitions and state representation as well as on the capabilities of efficient, deep, multi-agent planning. Specifically, we provide the following key contributions:

- An approach to simulation for emergent narrative, in which the domain description defines agent behaviors through **simple declarative actions and utility functions** over an **expressive and flexible state description** building on modern programming languages.
- A **planning model** that fulfills the **theory of mind** through the use of multi-agent **Monte Carlo Tree Search** on **partial worldviews**, using an efficient **data structure** for representing **local states** during planning.
- In a simple test scenario, an **experimental validation of the emergent behaviors** stemming from the model.
- A **delineation of the necessary next steps** to scale up the approach towards realistic scenarios.

Related Work

In her seminal 1999 paper, Aylett introduces the notion of *emergent narrative* and outlines its requirements and risks (Aylett 1999). She observes that emergent narrative is closer to experiencing real-world events than classic scripted storytelling. The challenge is that the simulation needs to "produce narrative often enough and with enough complexity to satisfy the user." Aylett asserts that agents should be continuously stimulated, independently of the observer's position and should exhibit a sufficiently-rich set of behaviors, preferably based on emotions.

Riedl and Bulitko's taxonomy on interactive narrative systems allows us to situate emergent narrative within the larger context of procedual storytelling (Riedl and Bulitko 2013). Two of the taxonomy's axes are decisive for classifying emergent narrative: *authorial intent*, which indicates the extent to which a human author determines narrative, and *virtual character autonomy*, which specifies the degree of independence that agents have from a central narrative system. On the axis of authorial intent, emergent narrative strongly lies at the *automatically generated* end, indicating that the resulting narrative is mainly created by the system while the human author has little influence on its outcome. On the axis for virtual character autonomy, emergent narrative lies on the *strong autonomy* side, as there is little centralized coordination of the agent's behavior.

Related work has explored wider regions of these axes. Noteably, Mateas, Stern and colleagues have produced a large body of work in the field of agent-based storytelling, including ABL, a declarative authoring language for agents (Mateas and Stern 2002a), and the game Façade, which was published in 2003 (Mateas and Stern 2003). Based on previous work (Bates 1992; Weyhrauch 1997), they have also extensively investigated *drama managers* (Mateas and Stern 2002b), subsystems that guide the simulation in order to achieve a certain narrative outcome. Often using planning to orchestrate agent behavior, drama managers have been central to a multitude of approaches to agent-based storytelling, for instance using partial-order planning (Riedl and Young 2010) or Monte Carlo Tree Search (MCTS)-based planning (Braunschweiler et al. 2018).

Recently, Ryan has criticized the use of intervening systems, such as drama managers, in emergent narrative (Ryan 2018). According to Ryan, imposing external control onto the simulation deprives it of true emergence and therefore does not result in true emergent narrative. We believe that this constraint is more consistent with Aylett's original definition and we have hence adopted it for this paper. Ryan has also laid out key challenges for future emergent narrative research (Ryan, Mateas, and Wardrip-Fruin 2015), which motivated the conceptual separation of simulation and curation (Ryan 2018).

Many existing implementations of agent-based storytelling use complex state representations and action definitions in order to achieve compelling agent interactions. Such approaches include simulating agents' emotional states (Aylett et al. 2006), using organizational models to guide agent behavior (Alvarez-Napagao et al. 2012), and simulating friendships (Ryan, Mateas, and Wardrip-Fruin 2016). McCoy and colleagues "Comme il Faut" system tracks relationships and agents' knowledge in order to simulate complex social behavior (McCoy et al. 2010), an approach which led to the game Prom Week (McCoy et al. 2012). Some approaches also explore the representation and simulation of agents' beliefs (Swartjes 2010), including allowing for lying and misremembering agents (Ryan et al. 2015), as well as extending the belief-desire-intention architecture for the narrative context (Berov 2017). By holding beliefs on other agents, an agent can reason on the knowledge and actions of its counterparts, leading to a the-

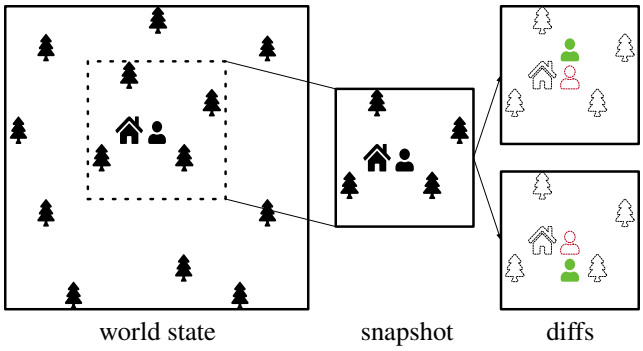

Figure 1: An overview of the *snapshot-diff* structure.

ory of mind. Reasoning while considering others has been achieved through performing agent actions in the belief state space (Shirvani, Ware, and Farrell 2017) and using dynamic epistemic logic (Eger and Martens 2017). Other approaches to theory of mind in agent simulation include the use of fixed-horizon exhaustive search and recursive models of other agents (Pynadath and Marsella 2005), and the simulation of mental influence (Chang and Soo 2008).

In contrast, we aim at achieving similar complexity through efficient deep planning, relieving us from the need of complex state representations and action definitions. To pick an action, we search the action space using MCTS. This method was originally developed to solve Markov Decision Processes with a large state space and a relatively small action space (Chang et al. 2005). In the context of emergent narrative, agents face similar conditions.

MCTS has been highly successful in building strong game AI, the most noteworthy example being the game of Go (Silver et al. 2017). In the field of story generation, Kartal and colleagues have shown the superiority of MCTS over other search algorithms (Kartal, Koenig, and Guy 2013). Jaschek and colleagues have used MCTS to explicitly reason about trust and distrust for the generation of murder mystery stories (Jaschek et al. 2019). In the context of interactive games, Sanselone and colleagues have used MCTS to allow non-playing characters to react to the player's action (Sanselone et al. 2014).

All these works have demonstrated the versatility and the power of MCTS as an algorithm. However, to our knowledge, there is no previous study that applies MCTS to emergent narrative with both algorithmic and computational performance in mind. Nor are we aware of any previous approach to emergent narrative in which theory of mind emerges from the planning process rather than from explicitly defined belief representations. This paper studies these aspects.

Model

Conceptual model We aim at designing a system that is generic, flexible and extendable. Following the principle of emergent narrative, our system simulates a set of individual agents within a world. Agents are defined in a declarative way, by providing a set of actions that each have a validity precondition and that affect the world's state when applied. Our system simulates each agent consecutively, with discrete time steps.

Keeping true to emergent narrative, our system performs planning on a per-agent basis without a global coordinator. Therefore, to choose which action to perform, each agent executes a full planning pass at each time step. While planning, each agent optimizes its own utility function, whose value depends on the state. The agents plan on the world state space as opposed to some narrative space, as our work addresses the simulation layer and we therefore do not enforce any narrative structure. Due to the depth of the search and the resulting combinatorial explosion, it is imperative that we use an adaptive search strategy (Kartal, Koenig, and Guy 2013). We thus developed a custom implementation of the MCTS algorithm (Chang et al. 2005).

While planning, each node in the search tree corresponds to a potential world state. To reduce state space explosion, each state only exists once at a given planning depth. This means that the same node can be reached through different action sequences (e.g. left then down and down then left), turning the tree into a more general directed acyclic graph. To compute node values, we follow (Chang et al. 2005), including the use of a discount factor γ. This factor causes the search algorithm to favor early reward over late reward, allowing MCTS to be used in non-terminating simulations such as ours. When reaching an unexpanded search node, we perform a rollout, i.e. simulating a sequence of random actions. Contrarily to the initial work (Chang et al. 2005), and similarly to recent works on board games (Silver et al. 2017), we do a rollout to the maximum depth for each leaf of the search graph.

In order to simulate limited knowledge and reduce computational complexity, each agent plans on a *snapshot* of the world state. It encompasses an agent's internal state as well as a subset of the external world state within a geometric or conceptual horizon of perception. If there are other agents within a subset of that horizon, the planning agent will also simulate their actions while planning. The actions of the other agents will be chosen so as to maximize their respective utility function, based on the planning agent's snapshot. Since the simulated actions of others are planned exclusively on the planning agent's knowledge of the world, they might diverge from the actual future actions of these agents. Conversely, the more congruent two agents' snapshots are, the more likely they are to correctly predict the counterpart's actions. By enabling each agent to infer the intentions of others based on its own perception, we endow the agent with a theory of mind.

To further reduce computational complexity and improve action flexibility, we introduce higher-level tasks whose execution last several time steps. For example, this allows moving to a specific location, instead of simply moving in a cardinal direction. While planning, an agent selecting such a task will commit to it until it either is completed or becomes invalidated due to changes in the planned state.

Software design Agents perform actions on the world state during execution and on snapshots while planning. To avoid copying these snapshots between each planning node,

we use a *diff* data structure which represents the divergence of a node's state from the root node's state (Figure 1). Formally, we define this data structure as follows: Let the world state W be a key-value map (e.g. a hashmap). Each key in W is unique and indexes either a primitive value or a nested key-value map (e.g., a 2-D map where each key is a coordinate and the value is a tile; or a map where each key is an agent identifier and the value a numerical property of that agent). An agent plans on a snapshot S, which consists of a subset of the keys in W with their corresponding associated values. During planning, agents perform actions that modify this snapshot S, but instead of modifying S directly, a diff D is created that holds a map of values for all modified keys (e.g., a list of updated tiles and changes in numerical properties). The state of a search node can hence be reconstructed by applying its diff D to the snapshot S.

Computational and memory efficiency is crucial to our approach. Our software design achieves this by storing search nodes in a hash table, indexed by the state of the node and the node's depth in the planning graph. This allows for fast graph traversal and node reuse during planning. To achieve maximum performance and compile-time memory safety, we choose Rust as the implementation language[2].

Experiments

Experimental setup

We run several experiments to validate the core algorithm and verify the emergence of complex and social behaviors from simple rules, without the need to explicitly encode these behaviors. Our experiments play out in a bounded 2-D tiled map, in which lumberjack agents collect wood. Each tile can be occupied by one object (Table 1, top). Each agent has its own inventory, which can carry an unlimited number of wood logs and one water bucket. Agents can only move through empty tiles and interact with objects on adjacent tiles. The actions that agents can perform vary from experiment to experiment, and are grouped by selectable features (Table 1, bottom). When planning, agents do not consider other agents as obstacles, but when performing actions, they do. Unless stated otherwise, our experiments use the following parameters:

- Discount factor $\gamma = 0.95$.
- Search depth $D = 45$ to ensure that an end action only has an effect of 10 % of the first one ($\gamma^D < 0.1$).
- A batch of 1000 visits per step.
- An agent only waits if no other action is applicable.
- Each experiment is repeated 100 times, average and standard deviation are shown in plots.

Validation of the algorithm

Effect of search depth In a first experiment, a single agent is placed on the left side of a corridor, and a single tree is placed on the right side, at a distance of 5 tiles (Figure 2, left). We vary the search depth with a number of visits exponentially proportional to the depth, in order to roughly cover

[2]https://www.rust-lang.org/

tree, height 1–3	rock	barrier	well	agents

Action	*Next to*	*Effect*
Base actions (always available)		
Walk (10)	*any*	Agent moves up, down, left or right by one step onto an empty tile.
Chop (20)	Tree	Reduces the height of the tree by 1 and provides the chopping agent with 1 wood. Chopping a tree with height 1 will remove the tree.
Waiting ( )		
Wait (1)	*any*	Agent does nothing. If no other action is possible, a *Wait* action is queued even if waiting is disabled.
Cooperation ( )		
Chop (20)	Tree	Same as *Chop*, with all adjacent agents receive one wood.
Blocking ( )		
Block (1)	*empty*	The agent uses one wood to build a barrier on an adjacent empty tile.
Watering ( )		
Pump (20)	Well	The agent gains one water bucket.
Water (20)	Tree	The agent loses one water bucket, adjacent tree grows to max. height.
Planting ( )		
Plant (1)	*empty*	The agent uses one wood to replace the adjacent empty tile with a tree with height 1.
Tasks ( )		
Walk to (10)	*any*	Agent moves onto an empty tile towards a given location.

Table 1: The possible tiles (top) and agent actions, with their respective weight and grouped by features (bottom).

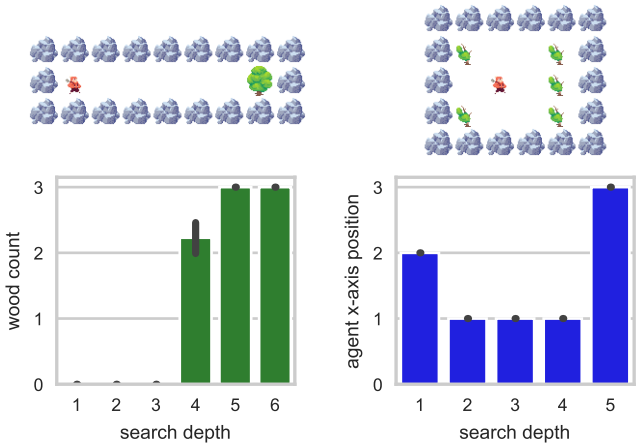

Figure 2: The effect of search depth. Left: The amount of wood collected after 10 turns. Right: The x position of the agent after 1 turn. Visit count = $100 \cdot 2^{D-1}$ where D is the search depth. Features: none.

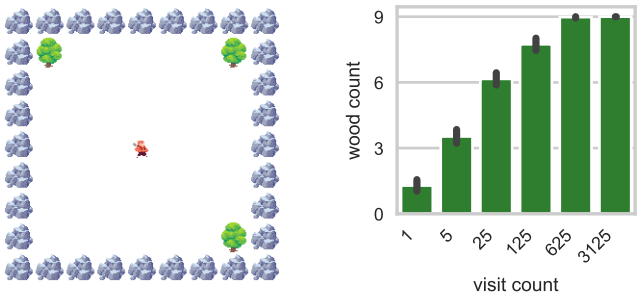

Figure 3: The effect of visit count. The amount of wood collected after 30 turns. Features: none.

a similar percentage of the search space independently of the tested depth. At depths 1–3, no wood is collected, as the agent does not find a course of action that leads to the tree. At depth 4, some wood is collected because, although the tree is initially beyond the search horizon, in some cases the agent moves right in the first steps, letting the tree enter the search horizon. At depths 5–6, the wood is fully collected as the agent first moves right as its only valid action, putting the tree within the search horizon.

In a second experiment, a single agent is placed near the center of a room with 2 trees on the left, and 3 trees on the right, but one step farther away (Figure 2, right). At depth 1, the agent does not move, because it does not find any tree. At depths 2–4, the agent first moves left because it sees the 2 trees. At depth 5, the agent first moves right, despite the trees being farther away, as it realizes that there is more wood to collect there.

Effect of visit count A single agent is placed in the center of a 7×7 area, with trees placed in three of the corners (Figure 3). We vary the number of visits. The average amount of wood collected, out of a maximum of 9, scales with the number of visits, reaching a virtually-perfect planning at 600.

Validation of emergence

Competition By considering others' actions in their plan, agents are endowed with a theory of mind. To validate that, we place two agents in a small arena with three trees of height 1 (Figure 4, top). One agent is next to all three trees, while the other is only adjacent to one. When planning for others is disabled, the first agent collects 2 wood. Most of the time, it starts by chopping either the top or the bottom tree, as these do not expand the search space as much as cutting the right one, and are hence favoured by MCTS. When planning for others is enabled, the first agent is able to exploit its advantageous position by chopping the tree closer to the other agent first, and hence collects 3 wood.

This ability is even more compelling when an agent can manipulate the world. To study that, we add the ability to place barriers, that is, to spend one wood to build an obstacle next to an existing one. When planning for others is disabled, the first agent can collect approximately 6 to 7 wood (Figure 4, bottom), out of a maximum of 12, since the other agent also collects wood. If planning for others is enabled,

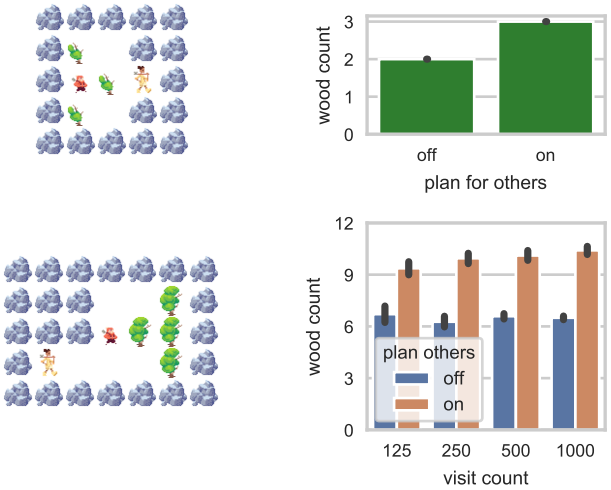

Figure 4: The emergence of competition. The amount of wood collected by the first agent (small red). Top: after 5 turns, features: none. Bottom: after 25 turns, features: , .

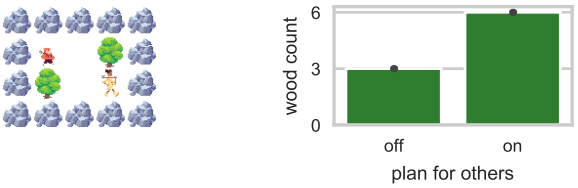

Figure 5: The emergence of cooperation. The amount of wood collected by the first agent after 10 turns. Feature:

the first agent can predict that the second agent will also cut wood, and therefore first chops the closest tree, invests that first wood to build a barrier, and then is free to collect all remaining wood. That is why, with sufficient visits, it can collect all 12 wood in most cases, and has 10–11 at the end (the barrier having cost 1 wood). This demonstrates the capacity of our agents to trade a short-term investment for a longer-term reward.

Cooperation Having a theory of mind can also enable cooperation. Figure 5 shows the amount of wood collected by the first agent, with and without considering the other's actions. Without it, each agent chops its own tree and receives 3 wood. With it enabled, they self-organize to chop trees at the same time, leading to 6 wood per agent. Note that the synchronization happens without any communication between the agents, but solely through their ability to reason about the other's moves.

Sustainability By allowing agents to water and plant trees, we study the ability of our planning system to create sustainable and efficient behaviors. An agent is placed in a corridor between a well on the left, and a tree on the right (Figure 6). To investigate the ability of an agent to predict the future, we vary the discount factor γ from 0.1 to 1.0. We vary the search depth accordingly, except for $\gamma = 1.0$, in which case we fix it to 150. When planting is disabled, with

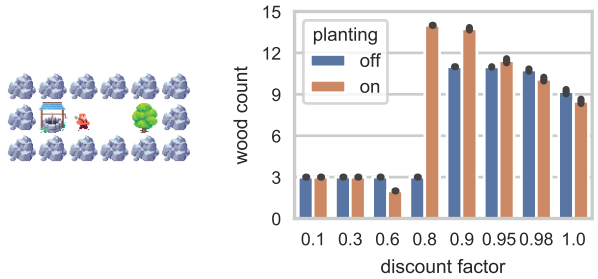

Figure 6: The emergence of sustainability. The amount of wood collected after 30 turns, with planting disabled (feature:) and enabled (features: ,).

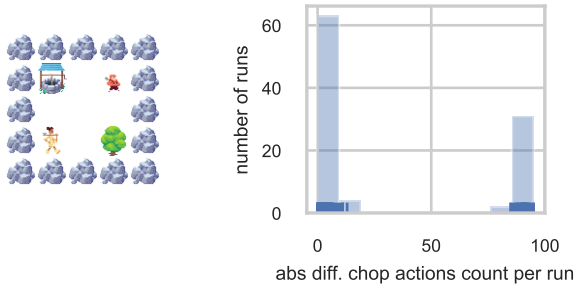

Figure 7: Emergence of agent specialization. Histogram of the difference in the number of chop actions between the two agents after 100 turns. Features: , ,

$\gamma < 0.9$, the agent greedily collects 3 wood, removing the tree. With $0.9 \leq \gamma \leq 0.98$, the agent is able to plan ahead, which creates a sustainable watering-chopping cycle. As γ approaches 1.0, the performance degrades because the agent has no incentive to take immediate action and hence loses time wandering around.

When planting is enabled, as soon as γ reaches 0.8, the agent is able to not only create a sustainable cycle, but also optimize that cycle by cutting the tree and replanting it on a closer tile. This leads to more wood over the 30 turns of the experiment. When γ reaches 1.0, the performance degrades even more than without planting, since the agent has more action options.

Agent specialization We study how agents can organize themselves to perform better than when acting alone. Two agents are placed symmetrically in a small arena (Figure 7, left). We see that in about 30 % of the cases, there is an absolute difference of 80–90 chop actions between the two agents (Figure 7, right). That indicates that one agent specializes in chopping while the other specializes in watering the tree. This shows that our system leads to the emergence of division of labour without explicit communication, simply by planning for others.

Task-based planning Agents' movements are planned using MCTS, which is an inefficient way to solve path planning. Figure 8 studies a simple environment in which an agent can collect an unbounded number of wood by continuously re-

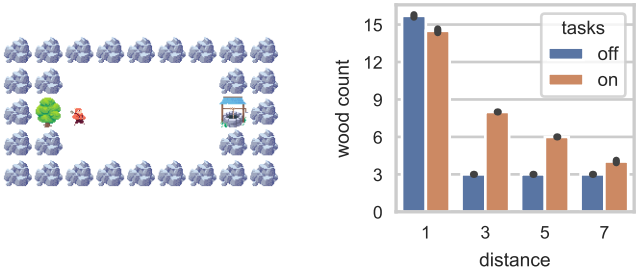

Figure 8: The effect of high-level tasks. The amount of wood collected after 30 turns, for different distances between the tree and the well (5 in map on the left), with normal (feature:) and task-based movements (features: ,).

growing the tree with water. For the default condition (tasks off), we can observe this behavior when the tree and the well are next to each other. But as we increase the distance between the well and the tree, this behavior disappears. This is due to the combinatorial explosion of possible paths back and forth from the well, making this option too hard to plan for. Therefore, the agent fully chops the tree and is left with 3 wood. Task-based planning solves this problem by planning with movement tasks that span multiple turns. The paths of these movement tasks are determined by an ad hoc A* path planner (Hart, Nilsson, and Raphael 1968). When these tasks are enabled, the agent never fully cuts down the tree, allowing for a sustainable behavior cycle. Though with increasing distance, the time spent travelling will decrease the efficiency of wood collection.

Discussion

The results of our experiments show that our approach can produce non-trivial behaviors based on a limited set of simple rules. These individual behaviors could be used as building blocks which, combined, could produce larger simulation systems. Furthermore, these emerging behaviors exhibit a complexity, especially at the social level, that exceeds that of the underlying rule system. This emergence of complexity is key for large-scale simulations in emergent narrative.

Lessons learned Our approach of unit-testing each feature proved to be key to validate the correctness of the implementation despite the inherent difficulty of debugging such a complex algorithm. Thereby, we observed an unexpected emergent behavior: Sometimes agents place barriers to limit their own search space, trading an early loss against the risk of future losses. Reducing γ limits this effect.

Expected performance Our current prototype implementation runs agent simulations at speeds suitable for interactive applications. For instance, during each planning step in the agent specialization experiment (Figure 7), our system performed 1000 visits to a search depth of 45 with an average branching factor of 4.6. Based on a sample of 100 turns, our system took an average 432 ms for one planning step on an Intel i7 8700K at 3.7 GHz. The resulting search trees had an average 703 nodes and size of 440 kB in memory, while

diffs had an average size of 125 bytes.

Due to the use of a limited perceptual horizon (e.g. neighboring or known agents), the combinatorial explosion is limited to that horizon. Thus, our approach has the potential to scale linearly with the total number of simulated agents in terms of computational requirements, and even superlinearly in terms of resulting social complexity (Helbing 2012). However, combinatorial explosion remains a limiting factor for an individual agent's intelligence, and future work shall focus on further curbing its impact.

A possible improvement is to project the actual state into a lower-dimensional representation prior to hashing. For instance, if another agent is far away, one could reduce its position to its quadrant and approximate distance. In that case, a node would become a projected state and hold a set of possible real states. When an action would be taken, the actual real state would be sampled. Hence, an action might lead to multiple successor states, as in (Chang et al. 2005).

Another optimization would be to learn the action value function. While we currently set it by hand, a first improvement could be to learn it per agent type, independently of the state. A second step would be to consider the snapshot state, for example through a deep neural network, as it is done in Alpha Go (Silver et al. 2017) or (Kartal, Hernandez-Leal, and Taylor 2019). This would greatly improve rollout performance, allowing for longer-term planning.

Potential for emergent narrative We believe that the burden of authoring the complex supporting logical structures that are necessary for producing rich multi-agent simulations represents the central limiting factor in the development of emergent narrative. Though our experiments themselves do not achieve emergent narrative, they do show that leveraging efficient planning combined with simple rules and per-agent state value functions can lead to the emergence of complex behaviors. This emergence demonstrates the potential to shift the burden of authorship from the human to the computer. As a result, our approach to simulation could lead to emergent narrative in large-scale story worlds, provided that the challenges of scalability and curation are solved.

Future work will need to develop a deeper understanding of how action variety, utility functions and map layouts affect the emergence of behaviors. For instance, abilities could be unevenly distributed among agents. Moreover, different utility functions could be used to endow individual agents with singular personalities.

On larger maps, factors such as the distribution of resources and the accessibility of different regions would determine the overall progression of the simulation. Future work should explore how inequality in agent placements, resource distribution gradients, islands and bottlenecks could create behaviors commonly seen in stories.

Our system is designed to be generic and extensible by letting users define the derivation of the snapshot from the world state. Future work should therefore also explore how to use the snapshot mechanism to implement perception, knowledge and beliefs. These could be modeled by changing the information that is passed from the world state to the snapshot, potentially sampling incomplete information. The snapshot could, for instance, include knowledge on previous states, whose granularity could depend on their recency. Beliefs could also be used within the utility function, requiring agents to first perceive and learn a counterpart's personality before being able to reliably predict its behavior.

Towards authoring and curation To make authoring easy and simple, future work should develop authoring tools for lay persons. These could embed front-end languages that are easier to write than Rust, high-level analytics that combine generic and user-defined metrics into high-level statistics, and easy debugging and tracing tools that allow to inspect the behavioral details of a single agent. Building on such tools, we envision a design methodology that relies on predictive global metrics.

To achieve true emergent narrative, future work will need to go beyond the simulation and address curation. The unpredictability of our bottom-up simulations will likely reveal new aspects of curation. Yet, we believe that accessible tools for simulation authoring provide excellent support for the study of curation, for example by facilitating psychological studies, which in turn could inform new metrics for simulation tuning.

Conclusion

In this paper, we laid out a novel path towards emergent narrative that relies on deep and efficient planning to achieve complex behaviors. We proved its validity in minimalist setups, in which competitive, collaborative and sustainable behaviors can emerge in a bottom-up simulation of simple agents. We identified the key scaling challenges for realistic scenarios, and proposed several refinements to address them. We believe that our exploratory work opens up new research opportunities towards large-scale simulations for emergent narrative.

Acknowledgements

We thank Violaine Fayolle for the lumberjack world art used in the visualizations, and Steven Poulakos for his input on related work. We also thank Lasse Lingens and Michael Keller who worked on earlier prototypes.

References

Alvarez-Napagao, S.; Gómez-Sebastià, I.; Panagiotidi, S.; Tejeda-Gómez, A.; Oliva, L.; and Vázquez-Salceda, J. 2012. Socially-Aware Emergent Narrative. In *Agents for Educational Games and Simulations*, 139–150. Springer.

Aylett, R.; Louchart, S.; Dias, J.; Paiva, A.; Vala, M.; Woods, S.; and Hall, L. 2006. Unscripted narrative for affectively driven characters. *IEEE Computer Graphics and Applications* 26(3):42–52.

Aylett, R. 1999. Narrative in virtual environments-towards emergent narrative. In *fall symposium on narrative intelligence*, 83–86. AAAI.

Bates, J. 1992. Virtual Reality, Art, and Entertainment. *Presence: Teleoperators and Virtual Environments* 1(1):133–138.

Berov, L. 2017. Steering Plot Through Personality and Affect: An Extended BDI Model of Fictional Characters. In *Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz)*, 293–299. Springer.

Braunschweiler, M.; Poulakos, S.; Kapadia, M.; and Sumner, R. W. 2018. A Two-Level Planning Framework for Mixed Reality Interactive Narratives with User Engagement. In *Artificial Intelligence and Virtual Reality (AIVR)*, 100–107. IEEE.

Chang, H.-M., and Soo, V.-W. 2008. Planning to Influence Other Characters in Agent-Based Narratives. In *Integrating Technologies for Interactive Stories Workshop, International Conference on Intelligent Technologies for Interactive Entertainment*, 12–17.

Chang, H. S.; Fu, M. C.; Hu, J.; and Marcus, S. I. 2005. An Adaptive Sampling Algorithm for Solving Markov Decision Processes. *Operations Research* 53(1):126–139.

Eger, M., and Martens, C. 2017. Practical Specification of Belief Manipulation in Games. In *Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*. AAAI.

Gardner, M. 1970. Mathematical games - the fantastic combinations of john conway's new solitaire game 'life'. *Scientific American* (223):120–123.

Grinblat, J., and Bucklew, C. B. 2017. Subverting historical cause & effect: generation of mythic biographies in Caves of Qud. In *Foundations of Digital Games (FDG)*, 1–7. ACM.

Hart, P. E.; Nilsson, N. J.; and Raphael, B. 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics* 4(2):100–107.

Helbing, D. 2012. *Social self-organization: Agent-based simulations and experiments to study emergent social behavior*. Springer.

Jaschek, C.; Beckmann, T.; Garcia, J. A.; and Raffe, W. L. 2019. Mysterious Murder - MCTS-driven Murder Mystery Generation. In *Conference on Games (CoG)*, 1–8. IEEE.

Kartal, B.; Hernandez-Leal, P.; and Taylor, M. E. 2019. Action Guidance with MCTS for Deep Reinforcement Learning. In *Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*, 153–159. AAAI.

Kartal, B.; Koenig, J.; and Guy, S. J. 2013. Generating Believable Stories in Large Domains. In *Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*. AAAI.

Mateas, M., and Stern, A. 2002a. A behavior language for story-based believable agents. *IEEE Intelligent Systems* 17(4):39–47.

Mateas, M., and Stern, A. 2002b. Towards Integrating Plot and Character for Interactive Drama. In *Socially Intelligent Agents: Creating Relationships with Computers and Robots*. Springer. 221–228.

Mateas, M., and Stern, A. 2003. Façade: An Experiment in Building a Fully-Realized Interactive Drama. In *Game developers conference*, volume 2, 4–8.

McCoy, J.; Treanor, M.; Samuel, B.; Tearse, B.; Mateas, M.; and Wardrip-Fruin, N. 2010. Authoring Game-based Interactive Narrative using Social Games and Comme il Faut. In *Electronic Literature Organization*, 1–8.

McCoy, J.; Treanor, M.; Samuel, B.; Reed, A. A.; Wardrip-Fruin, N.; and Mateas, M. 2012. Prom week. In *Foundations of Digital Games (FDG)*, 235–237. ACM.

Premack, D., and Woodruff, G. 1978. Does the chimpanzee have a theory of mind? *Behavioral and Brain Sciences* 1(4):515–526.

Pynadath, D. V., and Marsella, S. C. 2005. PsychSim: Modeling Theory of Mind with Decision-Theoretic Agents. In *International Joint Conference on Artificial Intelligence (IJCAI)*, volume 5, 1181–1186.

Riedl, M. O., and Bulitko, V. 2013. Interactive Narrative: An Intelligent Systems Approach. *AI Magazine* 34(1):67–77.

Riedl, M. O., and Young, R. M. 2010. Narrative Planning: Balancing Plot and Character. *Journal of Artificial Intelligence Research (JAIR)* 39:217–268.

Ryan, J. O.; Summerville, A.; Mateas, M.; and Wardrip-Fruin, N. 2015. Toward Characters Who Observe, Tell, Misremember, and Lie. In *Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*. AAAI.

Ryan, J. O.; Mateas, M.; and Wardrip-Fruin, N. 2015. Open Design Challenges for Interactive Emergent Narrative. In *Interactive Storytelling*, 14–26. Springer.

Ryan, J.; Mateas, M.; and Wardrip-Fruin, N. 2016. A Simple Method for Evolving Large Character Social Networks. In *International Joint Conference of DiGRA and FDG*. DiGRA.

Ryan, M.-L. 1987. The heuristics of automatic story generation. *Poetics* 16(6):505–534.

Ryan, J. 2018. *Curating Simulated Storyworlds*. Ph.D. Dissertation, University of California Santa Cruz.

Sanselone, M.; Sanchez, S.; Sanza, C.; Panzoli, D.; and Duthen, Y. 2014. Constrained control of non-playing characters using Monte Carlo Tree Search. In *Computational Intelligence and Games*, 1–8. IEEE.

Shirvani, A.; Ware, S. G.; and Farrell, R. 2017. A Possible Worlds Model of Belief for State-Space Narrative Planning. In *Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*. AAAI.

Short, T. X., and Adams, T. 2019. *Procedural Storytelling in Game Design*. A K Peters/CRC Press.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; Chen, Y.; Lillicrap, T.; Hui, F.; Sifre, L.; Driessche, G. v. d.; Graepel, T.; and Hassabis, D. 2017. Mastering the game of Go without human knowledge. *Nature* 550(7676):354–359.

Swartjes, I. 2010. *Whose Story Is It Anyway? How Improv Informs Agency and Authorship of Emergent Narrative*. Ph.D. Dissertation, University of Twente.

Weyhrauch, P. W. 1997. *Guiding interactive drama*. Ph.D. Dissertation, Carnegie Mellon University.