Computational Creativity in a Closed Game System

Cameron Browne and Simon Colton

***Abstract*—This paper describes the early stages of an experiment investigating the role of the computer as a creative collaborator in the game design process. We introduce the Shibumi set, a closed game system so simple that its rule space can be completely defined, yet deep enough to allow interesting games to emerge. Constraining the search space to such a closed system has computational benefits, but had unexpected effects on the creative process of designers during a related game design contest. These effects yield some insight into the creative process of experienced game designers, in particular, the way they search for rule sets to realise desired behaviours, and suggest a simple unified model of the game design process. We suggest ways in which these insights may be incorporated into future work, to produce software that might not only search for new games more effectively and assist the designer as a creative collaborator, but to automate the game design process in ways that might be perceived as more creative.**

I. Introduction

As *procedural content generation* (PCG) gathers momentum as a research topic [1], and automated systems start to invent board [2] and video [3], [4] games, questions of creativity arise. Is the combinatorial search for new games really creative? What role can software play in the design process?

In order to explore such questions, it is important to understand the design process used by humans designers when inventing games. One impediment to this understanding is that game design spaces are typically complex and not fully definable; we may attempt to model the design space for a given domain, but clever users will keep coming up with new rules and mechanisms that constantly expand this space. Fully defining such domains is not typically feasible.

In this paper, we describe a game system that is simple enough that its rule set can be fully defined, giving the computer the same access to the game design space as the human designer. The system is *closed* [5, p53] in that it is totally self-contained, and there is no exchange of rules or resources with its environment or other game systems.

We are interested in how the combinatorial advantage of the computer compares to the insight of the human designer, in finding interesting games within this closed space. Our primary research questions include:

1) *How can the computer be an effective collaborator, or even a peer, in the game design process?*
2) *How can we maximise the degree of creativity in the automated process?*

This closed game system constitutes the domain for an experiment in automated game design currently being conducted. The experiment is still in its early stages, but preliminary work has yielded insights into the game design process, which suggest approaches for improving the automated design process in future work. In particular, we attempt to locate *where* creativity lies in the game design process, and how to enhance it during automated search.

In Section II we describe the game system and a grammar for defining it. In Section III we describe a game design competition intended to generate a corpus of source games and to ensure that the rule space is complete. In Section IV we describe observations on the game design process and present a simple unified model. In Section V we describe practical ways in which this knowledge may be applied in our experiment, to improve the software's effectiveness as a creative collaborator, and (hopefully) increase the perception of the system as creative in the eyes of the user.

II. Domain

The ideal test domain for this experiment will be a game system simple enough that its entire rule set can be defined, but deep enough for interesting games to emerge. Specifically, the ideal domain will be:

- simple,
- non-trivial,
- unexplored,
- compact, and
- fully definable.

This section describes a turn-based, combinatorial board game system called the *Shibumi set*, designed specifically for this experiment with with these requirements in mind.

A. The Shibumi Set

The Shibumi set [6] consists of a square 4×4 grid of holes, and 16 white, black and red balls, as shown in Figure 1. 30 of these balls may be stacked to form a 4×4 *square pyramidal packing*. Games are by default two-player games between White and Black, in which red balls may be used as shared neutral balls, placeholders, scoring tokens, etc. The set is extremely simple while allowing games of sufficient depth to emerge, can be encoded efficiently, and its design space was almost entirely unexplored prior to this experiment.[1]

B. Efficiency

A full stacking will involve 4×4 + 3×3 + 2×2 + 1 = 30 balls. The state of each of these potentially playable points may be described by two bits, as follows:

- `00 = empty`

C. Browne and S. Colton are with the Computational Creativity Group, Department of Computing, Imperial College London, 180 Queens Gate, SW7 2AZ, UK.
E-mail: camb,sgc@doc.ic.ac.uk

[1] A known game, Pylos [7], is played with a subset of this equipment.

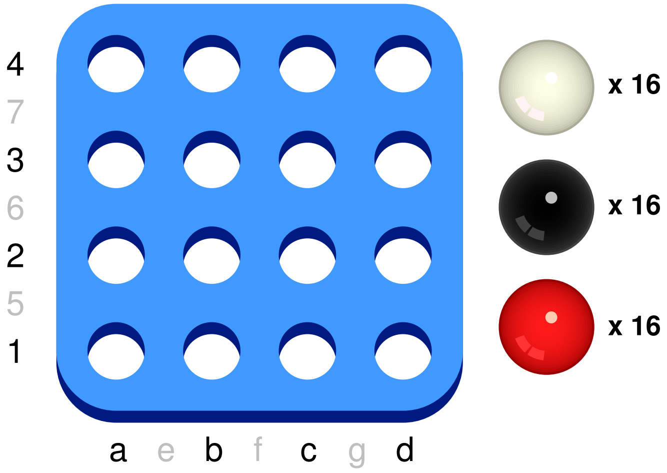

Fig. 1. The Shibumi set.

- `01 = white`
- `10 = black`
- `11 = red`

The complete game state requires 30×2 = 60 bits, which fits neatly into a single 64-bit `long`, with four bits left over to encode the current mover and winner. Encoding each board state as a single `long` has a number of advantages:[2]

- *Memory*: Smaller memory footprint.
- *Speed*: Less memory to be copied for state transitions.
- *Hashing*: Board state equals hash key (no encoding).
- *Forward Search*: Legal move lists can be stored as resulting board states rather than move descriptions, avoiding the need to re-apply moves during AI search.

Fig. 2 shows encodings of some typical starting positions. The empty board (`0x0`) is the default starting position. Some games start with a central 2×2 pyramid of red balls (`0x300003c3c00`), which occupy the pyramid's five interior points to guarantee that all further placements are at exterior points that will remain visible. Cases `0xaa000055` and `0x66996699` are typical starting positions for games that involve ball movement.

C. Shibui

The term *shibumi* comes from Japanese aesthetics, where *shibui* is the epitome of minimalist elegance [8]. Shibumi objects may initially look plain, but will reveal hidden depths the more time is spent with them. "Shibui" is the adjective that describes the overall concept, while "shibumi" is the noun that describes particular instances. Uppercase "Shibumi" refers to the game system in this paper.

This notion of simplicity hiding complexity is true of many abstract board games, as epitomised by the cliché: *a minute to learn, a lifetime to master*. However, it is especially true of the Shibumi set, given the discrepancy between its apparent 2D simplicity and actual 3D complexity. While the simplicity of the set implies a certain base level of elegance, the most highly prized Shibumi games are those in which the simplest rules produce the deepest and most interesting games.

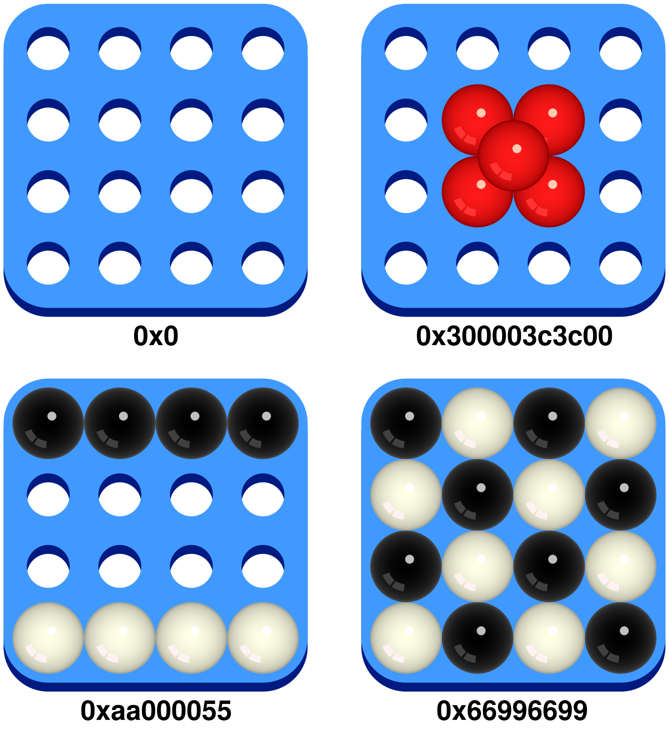

Fig. 2. Encodings (`long`) of some typical starting positions.

D. Complexity

The set has an upper bound on state space complexity of $2^{60} = 2,305,843,009,213,690,000$ states. However, only a fraction of this number will actually describe legal positions, according to the current rule set, and the fact that higher level placements must be supported by 2×2 platforms.

Of the 30 potential points, a maximum of 16 will be playable at any given time. This cap on the branching factor gives the set a move space complexity of a 2D game, but the state space complexity of a 3D game; the emergent possibilities may not be apparent on first glance.

Consider the game shown in Figure 3 (left), in which the mover must choose which colour to add at each turn. Any 2×2 platforms completed during the turn are capped by a ball of the minimum platform colour, if any, otherwise a ball of the maximum platform colour. The colour of the final apex ball is the winner. What colour should White play?

White only has three possible move choices {*white*, *black*, *red*} at the remaining playable point, but the outcome of each move can be hard to visualise. The correct move is to add a black ball for the win (Figure 3, right), as any other move results in a red apex ball. This game is relatively short (16 moves every time), but each of these moves can be difficult to evaluate correctly.

[2] Note that games involving additional information such as scores or ball counts will require another long to encode those values, and games involving the *ko* rule will require yet another long to record the previous board state.

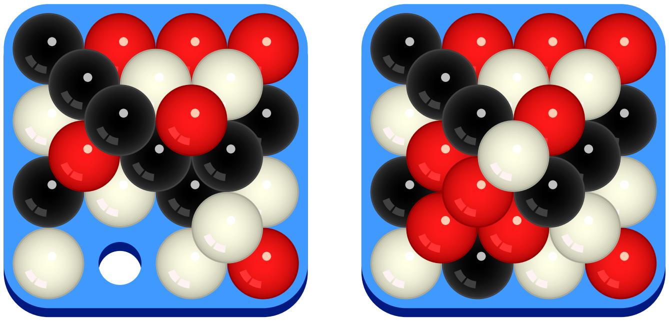

Fig. 3. What colour should White play?

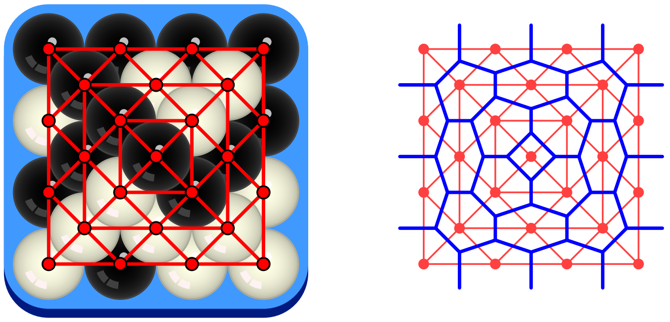

Fig. 4. Connectivity graph of a complete packing (left) and its dual (right).

Shibumi games can be more difficult for human players than mathematical analysis might suggest, due to the inherent difficulty of spatial visualisation. While such lack of clarity may reduce the strategic depth of some games, moves can often throw up some surprises, which is what keeps a game interesting for many players. This effective depth due to the opacity of 3D play limits the potential for strategic planning, as the clarity of a game dictates how far the player can see down the game tree [9]. Hence even a sub-optimal AI could provide players with sufficient competition.

It should be pointed out that depth in this context is relative. Shibumi games will not compete with Chess or Go in terms of depth, but can fare surprisingly well compared to other games of similar size. It is the unexpected return on depth for apparent simplicity that makes them shibui.

Most Shibumi games will be simple enough that their complete game tree can be expanded to *strongly solve* games in a reasonable time. For example, Pylos was solved with a day of computation and a 30 GB database [10]. However, this is not a serious drawback; human players will not be able to memorise such solutions – they are enumerations, not strategies – and it would not be feasible for a computer player to maintain and access such a database for *every* move of *every* Shibumi game. Full game tree analysis will be time consuming and should only be applied to the most promising candidates, but could give unparalleled insight into the nature of a game.

A feature of the Shibumi set is that the dual of the connectivity graph of the visible balls in a complete packing is *trivalent*, allowing deadlock-free connection games on the pyramid surface [11]. Figure 4 shows the connectivity graph (left) and its trivalent dual (right).[3]

E. Rules

The following ball locations are defined:

- `point`: The 30 potentially playable points.
- `hand`: Balls held in hand by each player.
- `pool`: Resource pool of balls, either controlled by one player or shared by all players.

Every move in the system comes down to shifting a ball from one point, hand or pool to another. However, the following move types are defined in the interests of convenience, comprehensibility and efficiency:

- `add`: Add a ball to a playable point, hand or pool.
- `remove`: Remove a ball from a point, hand or pool.
- `move`: Move a ball from one point/hand/pool to another.
- `push`: Push a ball along a higher level row or column, possibly pushing other balls, possibly to drop.

Various physical constraints on movement are enforced by the move generator, for example:

- Balls can only stack on 2x2 platforms.
- Balls that support two or more above cannot be moved.
- Removals may cause drops.
- Push drops must be deterministic, and so on.

For example, Figure 5 shows a game called Splush,[4] in which players may remove a ball of their colour, then must push a ball of their colour onto a higher level each turn, with the aim of forming all balls of their colour into a single visibly connected group. The move shown is the only winning move for White; the removal causes a drop that frees up a higher level point, allowing the winning push.

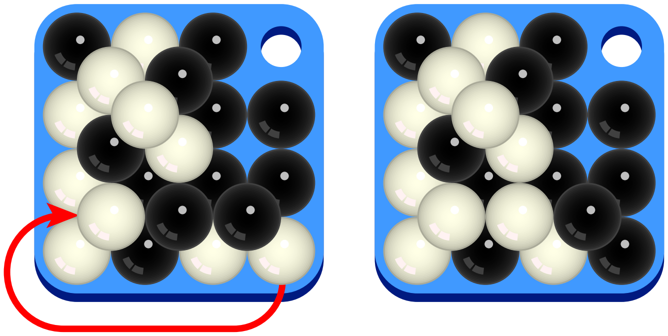

Fig. 5. This drop and push wins for White.

The following end conditions can trigger a terminal state, decide the result {`win`, `lose`, `draw`}, or both:

- `no-moves`: No legal moves.
- `complete`: Place the apex ball of the pyramid.
- `line`: Complete the specified line.

[3] This dual is coincidentally equivalent to the board design of the excellent connection game ConHex.

[4] Shibumi game names start with "Sp" to denote them as **S**quare **P**yramidal games; even the name space is constrained.

- `pattern`: Complete a pattern other than a line.
- `capture`: Capture the specified ball(s).
- `eliminate`: Eliminate the specified ball(s).
- `reach`: Reach the specified goal.
- `connect`: Complete the specified connection.
- `count`: Achieve the specified score or count.

This set of rules was found to allow a sufficiently wide range of deterministic, perfect information games that can be played with the set. Other rules may of course be devised by going to extremes, but the above set is complete within reason, and allows sufficient variety and depth. Note that the Shibumi system is truly closed in both the physical and abstract sense. No additional resources can be used, and additional rules cannot be borrowed from other game systems; nothing comes in or goes out.

F. Grammar

The system uses a grammar-based approach, in which the Shibumi rule space is defined in a standard *Backhus-Naur Form* (BNF) grammar. The first production rule indicates the basic form of a game:

```
<game> ::= (game <name> [<meta>]
                  <resources> <rules>)
```

Each game definition is a string parsable according to this grammar. For example, the following definition describes a game called Spline:

```
(game "Spline"
  (end win (line full flat any))
)
```

By default, two players (White and Black), take turns adding a piece of their colour to an empty playable point. Only the end condition needs to be specified in this case; a player wins by making a full line of their colour, in any flat direction. An elegant feature of Spline is that every game *must* be won before the apex ball is placed.

Sub-moves may be *tagged* (e.g. `m1`, `m2`, `m3`, ...), so that useful mechanisms can be chunked into *modules*, aiding comprehensibility and facilitating the crossover of meaningful mechanisms between games [12]. The following example shows a more complex move type for the game Splastwo:

```
(game "Splastwo"
  (play (turn (and m1 m2 m3))
    (m1 add (line flat ortho))
    (m2 (if (full (level to)) (inc score)))
    (m3 (if (= to apex) (inc score)))
  )
  (end win (>= score 3))
)
```

In this game, players take turns adding a flat orthogonal line of their colour, of any length. 1 point is scored if the move fills (completes) a level, and 1 bonus point is scored for completing the apex. A player with no legal moves (i.e. no remaining balls) must pass by default. The game is won by the player who scores 3 or more points; again, every game must produce a winner.

III. Shibumi Challenge

In order to generate a number of source games for the experiment, a game design contest called the Shibumi Challenge was held over December 2011 and January 2012. The aims of this contest were to:

1) Produce a corpus of source games to seed the search.
2) Ensure that our closed rule set was complete.
3) Observe the game design process in action.

The Challenge was run through the BoardGameGeek (BGG) web site, and attracted 45 entries from 22 designers [13], [14]. Entries were playtested and scored by the organisers (Cameron Browne, Néstor Romeral Andrés and Stephen Tavener) according to the following criteria:

- *Clarity*: Rules should be simple and easily understood.
- *Depth*: The game should be interesting and replayable.
- *Fun*: How much we simply enjoyed each game.
- *Elegance*: How harmoniously the rules combined.
- *Popularity*: Number of votes received by onlookers.
- *Originality*: Novelty of rules and mechanisms.
- *Usage*: How well the simple equipment was used.

Prizes were awarded to the three highest scoring games:

1) *Sploof* by Matt Green.
2) *Spire* by Dieter Stein.
3) *Splastwo* by Giacomo Galimberti (listed in II-F).
4) *Sprite* by Micah Fuller (runner-up).

The challenge entries included several novel rules and mechanisms, and the general quality was sufficient to warrant the publication of a *Shibumi Rule Book* [15]. The Challenge served its intended purpose, as a form of crowd-sourcing to generate source games and maximise coverage of the rule space. The Shibumi set has been studied in enough depth, by enough designers, for us to be reasonably confident that the most important rules for the set are now known. These will provide the raw material for the upcoming search.

Despite the success of the Challenge, the organisers were struck by two effects that became apparent during its course:

1) Creating new and interesting Shibumi games is hard.
2) Some experienced designers refused to participate.

The first effect was not a surprise; creating new games is easy, but creating interesting ones from such simple components is not. As mathematician Ian Stewart points out:

> *The simpler the ingredients, the harder it is to make things with them.* [16, p52]

The second effect was much more surprising. Several experienced game designers took an interest in the Challenge, but flatly stated that they would not participate. Some had previously participated in an even more minimalist game design contest,[5] so we were interested in why they baulked at this apparently simple challenge. We believe that both of these

[5] The self-explanatory “4 cards or tiles only” game design contest: http://www.boardgamegeek.com/thread/383408/

effects can be explained with some insight into the creative processes involved.

IV. The Game Design Process

We now examine the creative processes observed in use by game designers, and the creative spaces in which they operate, in order to explain these effects and to maximise the possibility of creativity in the automated design process. Such analysis is more tractable in a closed system such as the Shibumi set.

A. Design Spaces

Boden defines creativity as the ability to come up with artefacts that are *new, surprising and valuable* [17, p1]. She describes two basic forms of creativity:[6]

1) *Transformational*: The creation of new search spaces.
2) *Exploratory*: The exploration of these search spaces.

The possibility of transformational creativity is removed in the current experiment, as the system is closed; any creativity must come from the exploration of combinations of the precisely defined rule set, rather than from external sources. The exploratory mode resonates strongly with the Platonic view of mathematics [18]; the games are all there in the search space waiting to be found, just as Michelangelo observes that any conceivable sculpture is right there in the marble [19, p.167].

In the following sections, we now consider two apparently conflicting approaches used by experienced game designers within these creative spaces, and how they can be related to provide a simple unified model of the game design process.

B. Organic Design

Christian Freeling, the inventor of Havannah [11], stated that he would not participate in the Shibumi Challenge, as its format was not conducive to his approach to game design. He describes this as *organic*; the game initially has no form, but grows in the designer's mind as a theoretical organism that "moves or grabs or hunts in a particular way". The designer must then experiment to find the combination of rules that most closely realises this desired behaviour:

> *A mechanism becomes 'organic' if it naturally points to a theme that fits its behaviour, its will and intent.* [14]

Freeling saw the closed environment of the Shibumi set as a barrier to this process, stating a preference for asking a game for its rules rather than exploring rules for their games. This view places creativity squarely in the transformational realm.

C. Serendipitous Design

Another approach to game design is described by leading puzzle designer Raf Peeters [20]. He emphasises the importance of *serendipity* in his approach, which can be summarised as:

1) *Search*: Actively searching within the design space.
2) *Insight*: Realising when a new combination is good.

Invention does not start with "one big idea but with hundreds of small ideas" [20]. The key is to realise when a serendipitous meeting of ideas yields something of value, which can often occur when the designer least expects it and is not even thinking about the problem.

Peeters suggests that the most valuable Shibumi games will be those with novel mechanisms that transcend the set and have broader application to other domains [21]. Further, he points out that designers might have baulked at the Shibumi Challenge not because of the simplicity of the equipment, but because the domain was closed; designers were not allowed to "think outside the box". Rather than focussing on what is possible, they may have focussed instead on what is not, seeing the limiting 2D appearance of the board rather than the emergent 3D reality of the set.

D. Unified Model

These two approaches superficially appear diametrically opposed: in the organic approach the behaviour disctates the rules, while in the serendipitous approach the rules dictate the behaviour. However, Fig. 6 shows an alternative interpretation that reconciles both into a single unified model.

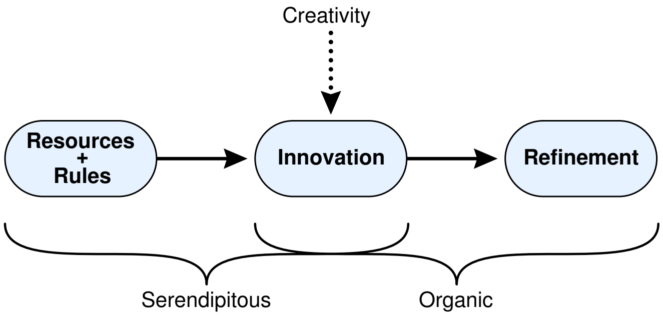

Fig. 6. Unified model of the game design process.

In the unified model, the designer seeks innovation in the available material (i.e. the resources and rules defined by the system), then refines their combination to realise the desired innovation. The act of innovation correlates with Peeters' serendipitous approach, while the act of refinement correlates with Freeling's organic approach. Note that this can be a recursive process if the refinement step requires further innovation to succeed.

The creativity within this process lies primarily in the innovation of new ideas. The designer must immerse themself in the design space and seek harmonious and novel innovations; the more harmonious and the more novel, the more valuable. The refinement step is more one of fine tuning, which may involve some degree of creativity to solve immediate problems, but is more a process of observation, trial and error.

For example, Challenge participant Giacomo Galimberti had the innovative idea to devise a Shibumi game with Nim-like behaviour, and pursued this goal through a number of refinements, including Special Cake and Splast (neither of which worked as intended) until its final incarnation as the bronze medal winner Splastwo, listed in Section II-F. Splastwo

[6] Boden actually proposes three forms of creativity, but the third – combinatorial – is really a type of exploration.

provides a novel twist on the genre of Nim-like games, and the 3D geometry of the set integrates well with a simple reduction mechanism to produce a result that is greater than the sum of its parts. Being a Nim-like game it would seem that there should be a simple winning strategy, although this is complicated by the stacking mechanism and limited piece count, and continues to elude discovery.

Professional game designer and publisher Mike McManaway describes a similar process behind the invention of his flagship product Tantrix [23]. His innovation was to devise a path completion game on hexagonal tiles, with the additional constraints that the final game should be simple, attractive, and highly marketable.[7] The rules underwent many refinements over several years before settling on the current successful format.

E. Creative Flow

This unified model has many similarities with the four-point theory of creative thought proposed by Wallas [24]:

1) *Preparation*: Familiarity with the available material.
2) *Incubation*: Immersion in the design space.
3) *Illumination*: The "eureka!" moment of innovation.
4) *Verification*: The nuts-and-bolts of realising the idea.

Smith describes the harmful effect on the incubation stage of *fixation* on known schema, and points to the "counterproductive effect of prior experience" [25, p.239]. Thus the designer may need to get out of a "mental rut" for incubation and innovation to occur.

In terms of game design, this may be achieved with the flow of ideas across domain boundaries, so that known ideas get tried in new contexts. Peeters sees creativity as lying at the borders where different domains meet and the exchange of ideas across these borders, and values most highly those inventions that surprise him [21]. This is similar to Hofstadter's notion of *slippage* between schemas, which he describes as "the very crux of fluid thought" which leads to creativity [22, p.237].

Such flow is not possible to achieve across the boundary of our closed system, but may be possible to achieve across conceptual boundaries of sub-domains *within* the system. Such sub-domains might include capture games, N-in-a-row games, completion games, race games, and so on, all defined within the Shibumi system, and whose components can be recombined in novel ways.

We argue that there is still room for creativity within the closed system, although the designer may have to work harder to achieve it. This could require a more careful and thorough exploration of the search space, increasing the chance of finding unexpected hidden gems.

F. Creativity Within Constraints

We further argue that the more constrained the system, the more creativity is required to achieve meaningful results within it; the constraints imposed by the Shibumi set do not quash creativity, but in fact evoke *more* creativity from designers. This view is shared by most game designers to which this question was put.

Haught and Johnson-Laird even go so far as to say that the need for "freedom to create" is an illusion, and that "without constraints, there is no creativity" [26]. According to Stravinsky:

> *The more constraints one imposes, the more one frees one's self.* [27]

Game designers Russ Williams [28] and Giacomo Galimberti [13] independently drew an analogy between the Shibumi set and the haiku. The haiku author must find expression with a handful of words that conform to a very rigid structure. But this does not mean that the haiku allows less creativity than freer forms of expression such as the novel; if anything, *more* creativity is required to make the most of this constrained system.

Constraints on design can actually be liberating once they are accepted; the task is more clearly defined, and all energy can then be focussed on achieving the best results from what is available. Maeda advises a philosophy of "thoughtful reduction" in design, to remove the unnecessary and hide complexity where possible: "with more constraints, better designs are revealed" [29, p97]. Constraints focus the search on smaller and more relevant parts of the design space [30].

V. Practical Implications

We now consider some practical implications of these observations on the game design process, and the creative spaces involved, for enhancing creativity within a closed system such as the Shibumi set.

A. Explaining the Effects

The observations on the game design process listed above go some way to explaining the effects observed in Section III: that it is hard to invent new and interesting games for the Shibumi set, and that several experienced game designers simply baulked at the Shibumi Challenge.

We believe that these are both primarily due to the fact that the system is closed, removing any possibility of transformational creativity or exchange of ideas with other game systems; any creativity must come from innovative combinations of the resources and rules defined within the system. Designers can not rely on their intuition and experience with other game systems, but must work with the material at hand, and must be prepared to search harder to find truly innovative combinations within this limited space.

Further, if the designer is not prepared to study the system in sufficient depth to appreciate the full range of possibilities that it offers, then their options will seem even more limited; they might only see the 2D appearance of the board rather than the emergent 3D reality of the set. The Shibumi system will not support every mechanism or behaviour that a designer may imagine, based on their prior experience with other game systems, but it can still support a surprising number, given sufficient insight.

[7] Verified in conversation with McManaway.

B. Creative Search in a Closed System

One of the aims of this experiment is to explore the shape of the design landscape of a closed game system such as the Shibumi set. For this purpose, we will be comparing standard evolutionary search methods to Monte Carlo tree search (MCTS) methods [15], which we believe may offer some benefits in this respect [31]. We are particularly interested in whether the automated search will find hidden gems missed by human designers, the frequency with which these occur, and the margin by which they were missed.

Ritchie [32] describes three essential criteria that items produced by an automated system should satisfy for deciding whether creativity has occurred:

1) *Typicality*: Degree to which items represent the class.
2) *Quality*: Quality or utility of the items.
3) *Novelty*: Dissimilarity with existing items.

Typicality is not an issue here, as any game definable in the Shibumi grammar should be recognisable as a combinatorial game. Game quality can be estimated using aesthetic measurements of self-play trials between artificial agents, as demonstrated in [2], and used as a fitness function to guide the automated search within the Shibumi design space. In order to address the issue of novelty, we distinguish between two types of search according to the unified model shown in Fig. 6: *innovation search* and *refinement search*.

1) Innovation Search: Creativity may be enhanced in the automated search by seeking novel and unexpected innovations, especially ones that produce interesting emergent behaviour not evident in their component parts. We therefore wish to combine dissimilar ideas from the search space during the innovation stage, to maximise the transfer of rules between sub-domains within the closed system. This may be achieved by selecting individuals for recombination with probability based on their functional distance, so that games are mated more often with less similar games, and rules are recombined more often with less similar rules (i.e. rules that occur more often in less similar games).

To this end, a reliable distance metric between pairs of games is required. Gauging game distance by genotype (rule set) can be unreliable, so we will gauge game distance by phenotype (resulting play), by "reverse engineering" rule sets through simulated play, to identify shared behaviours, using a *functional game distance* metric.

2) Refinement Search: Once the innovation search has produced a (hopefully interesting) mechanism to explore, we wish to temporarily fixate on this rule combination and refine it with localised combinatorial search, to find the optimal combination of matching rules. This can be encouraged in a grammar-based approach by instantiating the desired rule(s) in the generating grammar.

For example, if the innovation is to combine "win with a line of size N" with "lose with a group of size M", then the grammar's default <end> rule:

```
<end> ::= (end <result>)
```

may be instantiated with the appropriate terminal symbols:

```
<end> ::= (end
            (win (in-a-row <int>))
            (lose (group <int>))
          )
```

The search will then continue for games that include this mechanism. Note that the line length and group size parameters remain as non-terminal <int>s, so that these values may be tuned by the search, in addition to the other rules of the game. MCTS is well suited to this kind of refinement search, as its decision choices are systematic and guided by a mechanism that balances *exploration* and *exploitation* with logarithmic regret [33].

If the system is to be a creative collaborator in the game design process, users should be allowed to instantiate their own innovations in lieu of conducting the innovation search stage. They could then test their own ideas for innovative rule combinations, but still harness the combinatorial power of automated search for the refinement stage. Educated users might specify their choices directly in the grammar, while novice users might simply specify preferred games to be selected for recombination. This would allow the user to drive the creative process, hopefully restoring some impression of transformational control, ironically by constraining the search space further.

C. The Perception of Creativity

So far we have discussed creativity in the design process from the perspective of the *producer* (i.e. designer). We now turn to the perception of creativity from the viewpoint of the *consumer* (i.e. player).

A significant hurdle to the perception of machine creativity is the vicious circle described by Colton [34]:

> *The default position that software is not creative leads to a low assessment of an artefact which it produces, but then if the software produces bad artefacts, it really cannot be creative.*

One advantage of combinatorial board games over other creative domains is that artefacts can be described in abstract terms, independent of theme, so that a rule set produced by machine is indistinguishable from a rule set hand-crafted by a human designer. It will evoke the same responses (i.e. strategies and moves) regardless of who created it. We could exploit this advantage by simply not telling players which games are automatically produced. In the end, this may not even be important; players were surprisingly accepting of the computer-generated game Yavalath even after its origins had been revealed [35]. But such deception would be a retrograde step from a computational creativity perspective, if we want to promote computers as creative agents in their own right.

Instead, full disclosure of the underlying design process is more likely to increase the perceived creativity of the generating system. This has the added advantage of giving automatically generated artefacts a "back story" that the consumer can latch onto. For example, the user might be interested to know that: "This game plays similarly to X and Y, but rule R was changed to S to avoid cycles and maximise tension".

1) Design Tutor: The system might communicate its design choices to the user through a *viva voce* process, as suggested by Alan Turing [36], in which the user queries the system to gain a deeper understanding of its choices. However, a more user-friendly system might simply volunteer such information: if a game is deemed unplayable then the system could specify *why* this is so, and if game A is preferred over game B then the system could specify *why* it prefers A.

The system could also suggest rule fixes to address observed shortcomings, or likely directions to take a given rule set, possibly serving a useful function to the designer as a game design tutor. It would probably be considered a creative act if the system suggested useful ideas that the designer had not yet considered.

VI. Conclusions

In designing software to act as a creative collaborator in the game design process, we wish to emulate the creative process used by human designers. This process is observed in the context of a simple, closed game system called the Shibumi set, and a game design contest conducted to generate a corpus of source games. While constraining the search to such a closed and completely defined system has computational benefits and levels the playing field, giving the software access to the same search space as the human designer, it also had a significant impact on the creative process itself. Insulating the system from the general design space precludes cross-fertilisation of ideas between this system and other game types, removing the transformational aspect of creativity and allowing only the exploratory.

However, we argue that there is still sufficient scope for true creativity to occur in this closed system. Studying the creative process of designers reveals apparently conflicting approaches that can be unified into a simple model of the game design process. We harness this knowledge to suggest ways in which the automated search may be adapted to more closely resemble the process used by designers, so that the system may search more effectively, operate as a useful collaborator in the game design process, and improve its chances of being perceived as creative in its own right.

Acknowledgements

The authors would like to thank Giacomo Galimberti, Don Kirkby, Nathan Morse, Raf Peeters and Néstor Romeral Andrés for useful discussions and suggestions. We would also like to thank the anonymous reviewers for their helpful suggestions. This work was supported by EPSRC grant EP/I001964/1 as part of the research project *UCT for Games and Beyond*.

References

[1] J. Togelius, G. N. Yannakakis, K. O. Stanley, and C. Browne, "Search-based Procedural Content Generation: A Taxonomy and Survey," *IEEE Trans. Comp. Intell. AI Games*, vol. 3, pp. 172–186, 2011.

[2] C. Browne, "Automatic Generation and Evaluation of Recombination Games," Ph.D. dissertation, QUT, Brisbane, 2008.

[3] A. M. Smith and M. Mateas, "Variations Forever: Flexibly generating rulesets from a sculptable design space of mini-games," in *Proc. IEEE Conf. Comput. Intell. Games*, 2010, pp. 273–280.

[4] M. Cook, S. Colton, and J. Gow, "Initial Results from Co-operative Co-evolution for Automated Platformer Design," in *Proc. Applic. Evol. Comput. (EvoGAMES)*. Springer, 2012, pp. 194–204.

[5] K. Salen and E. Zimmerman, *Rules of Play: Game Design Fundamentals*. Cambridge, Massachusetts: MIT Press, 2004.

[6] N. Romeral Andrés, "The Shibumi Set," 2011. [Online]. Available: http://www.nestorgames.com

[7] Gigamic, "Pylos Classic," 2011. [Online]. Available: http://www.gigamic.com/pylos-classic-c-31-p-31.html

[8] C. Browne, "Elegance in Game Design," *IEEE Trans. Comp. Intell. AI Games*, vol. 4.

[9] R. Abbott, "Under the Strategy Tree," *Games & Puzzles*, vol. 36, Addendum, 1975.

[10] O. Aichholzer, D. Detassis, T. Hackl, G. Steinbauer, and J. Thonhauser, "Playing Pylos with an autonomous robot," in *Proc. IEEE/RSJ Conf. Intell. Robots Syst.*, Taipei, 2010, pp. 2507–2508.

[11] C. Browne, *Connection Games: Variations on a Theme*. Natick, Massachusetts: AK Peters, 2005.

[12] L. Spector, K. Harrington, and T. Helmuth, "Tag-Based Modularity in Tree-Based Genetic Programming," in *Proc. Genet. Evol. Comput. Conf. (GECCO)*, Philadelphia, 2012, pp. 815–822.

[13] C. Browne, "The Shibumi Challenge," 2012. [Online]. Available: http://www.boardgamegeek.com/geeklist/75022

[14] ——, "AI Game Design: The Shibumi Challenge," 2012. [Online]. Available: http://www.boardgamegeek.com/blogpost/6931

[15] C. Browne and N. Romeral Andrés, *Shibumi Rule Book*. Raleigh: Lulu, 2012.

[16] I. Stewart, *Professor Stewart's Cabinet of Mathematical Curiosities*. London: Profile Books, 2008.

[17] M. A. Boden, *The Creative Mind: Myths and Mechanisms*, 2nd ed. London: Routledge, 2004.

[18] M. Balaguer, "Platonism in Metaphysics," in *Stanford Encyc. Phil.*, E. Zalta, Ed., 2004. [Online]. Available: http://plato.stanford.edu/archives/sum2009/entries/platonism/

[19] I. D'Israeli, *The Literary Character*. London, UK: John Murray, 1822, vol. 1.

[20] R. Peeters, "Inventing and Serendipity," 2011. [Online]. Available: http://www.smartgamesandpuzzles.com/inventor/Blog/Blog.html

[21] ——, "Personal correspondence," 2012.

[22] D. Hofstadter, *Metamagical Themas: Questing for the Essence of Mind and Pattern*. New York: Basic Books, 1985.

[23] M. McManaway, "Strategy game twenty years old," 2008. [Online]. Available: http://christchurchcitylibraries.com/entertainment/ people/m/mcmanawaymike

[24] G. Wallas, *The Art of Thought*. New York: Harcourt, 1926.

[25] S. M. Smith, *The nature of insight*. Cambridge: MIT Press, 1994, ch. Getting Into and Out of Mental Ruts: A Theory of Fixation, Incubation, and Insight, pp. 121–149.

[26] C. Haught and P. Johnson-Laird, "Creativity and Constraints: The Production of Novel Sentences," in *Proc. 25th Annual Conf. Cognit. Science Soc.* Mahwah: Erlbaum, 2003, pp. 528–532.

[27] I. Stravinsky, *Poetics of Music in the Form of Six Lessons*. Harvard: Harvard Univ. Press, 1947.

[28] R. Williams, "Shibumi as a game system," 2011. [Online]. Available: http://www.boardgamegeek.com/thread/721722

[29] J. Maeda, *The Laws of Simplicity*. Massachusetts: MIT Press, 2006.

[30] A. Liapis, G. N. Yannakakis, and J. Togelius, "Adapting Models of Visual Aesthetics for Personalized Content Creation," *IEEE Trans. Comp. Intell. AI Games*, vol. 4.

[31] C. Browne, "Towards MCTS for Creative Domains," in *Proc. Int. Conf. Comput. Creativity*, Mexico City, 2011, pp. 96–101.

[32] G. Ritchie, "Some Empirical Criteria for Attributing Creativity to a Computer Program," *Minds & Machines*, vol. 17, pp. 67–99, 2007.

[33] C. Browne, E. Powley, D. Whitehouse, S. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A Survey of Monte Carlo Tree Search Methods," *IEEE Trans. Comp. Intell. AI Games*, vol. 4:1, pp. 1–43, 2012.

[34] S. Colton, "Creativity Versus the Perception of Creativity in Computational Systems," in *Proc. AAAI Spring Symp.: Creative Intell. Syst.* AAAI, 2008, pp. 14–20.

[35] C. Browne, *Evolutionary Game Design*. Berlin: Springer, 2011.

[36] A. Turing, "Computing Machinery and Intelligence," *Mind*, vol. LIX, pp. 433–460, 1950.