Overcoming Algorithmic Bias as a Measure of Computational Creativity

Jonathan Demke and Dan Ventura
Computer Science Department
Brigham Young University
demkejon001@gmail.com, ventura@cs.byu.edu

Abstract

The field of computational creativity (CC) stands to benefit significantly from the establishment of a benchmark test to validate and compare CC systems. While previous efforts have identified criteria necessary for creating such a benchmark, we identify a crucial missing element: the ability to overcome bias. In human tests of creativity, a key focus lies in determining an individual's capacity to surmount cognitive biases. Similarly, we argue that CC systems must overcome algorithmic biases. To address this, we introduce definitions of bias and overcoming bias within CC systems, leveraging Wiggins' creative systems framework. Consequently, this raises questions about transformational creativity and how it is to be characterized.

It took me four years to paint like Raphael, but a lifetime to paint like a child. – Pablo Picasso

Introduction

The computational creativity (CC) community needs a creativity benchmark test to gauge the efficacy and advancement of CC systems. While there might be reservations about state-of-the-art chasing (Jordanous 2022) in the CC field, it is evident that benchmarking has driven progress in various realms of artificial intelligence (AI) research, exemplified by benchmarks such as Imagenet for computer vision and Atari for reinforcement learning (RL). Providing clear objectives has proven beneficial for many fields. However, the design of a benchmark test for CC presents unique challenges.

Evaluating CC systems is difficult both because the domains of interest are so varied and disparate, making the discovery of a unified evaluation approach difficult and because evaluating outcomes in said domains is often subjective, leading to disagreement on whether one system is "more creative" than another. Moreover, employing human evaluators for such assessments can be costly and time-consuming. Despite these challenges, we are optimistic regarding the feasibility of establishing a computational benchmark for evaluating creative systems.

Recent research provides compelling arguments regarding a potential methodology and criteria that might be useful in developing a benchmark test. Particularly noteworthy is the idea that testing within a creative domain with objective rules and evaluation metrics, as in the case of creative word games (Spendlove and Ventura 2022), significantly ameliorates the hurdle of subjectivity. Additionally, (Spendlove and Brown 2023) argues for certain criteria necessary for a task that requires creativity, one of which is the presence of a large meaningful state space that cannot be brute-force searched. While we agree with this sentiment, we believe that even toy problems with small state spaces can offer a valid mechanism for testing creativity under the right circumstances. Therefore, we propose a more general criterion as a hallmark of creativity, that of *overcoming bias*, which might also be referred to as "thinking outside the box".

A bias is a(n) (un)conscious decision that is "informed, either correctly or incorrectly, by some *a priori* belief or understanding we already possess" (Loughran 2022). Biases may be beneficial or detrimental, both providing heuristics for rapid decision-making and obfuscating the otherwise obvious.

It is commonly accepted that when a human overcomes a cognitive bias to reach a solution that diverges from their predispositions, they are engaging in a creative process, and notably, numerous assessments of human creativity center on discerning just such a capacity to overcome bias. We therefore advocate for the inclusion of this criterion in the evaluation of CC systems, asserting that for CC systems to engage in the creative process they too must overcome their (algorithmic) biases: *The ability to overcome bias is fundamental to the (computational) creative process.*

Subsequent sections will explain the rationale behind this claim; explore the effects of this perspective in the context of P- and H-creativity (Boden 1991); and extend the notion of overcoming bias to computational systems, utilizing Wiggins' creative systems framework (CSF) (Wiggins 2006).

Overcoming Bias

There is a difference between the ability to create an artefact that is deemed creative by society and the ability to engage in the creative process, and this distinction will be explored later when the role of bias on P- and

H-creativity is discussed. Our central thesis contends that the creative process involves overcoming bias. We support our claim by dissecting human tests of creativity, each of which tests the subject's ability to escape cognitive biases, whether implicitly or explicitly. Furthermore, we categorize tests of creativity in a way that underscores bias as a foundational element.

Human Tests of Creativity

There are many tests for human creativity: Duncker's Candle Problem (Duncker 1974), Guilford's Alternative Uses Task (Guilford 1967), Remote Associates Test (RAT) (Mednick 1962), Divergent Association Task (DAT) (Olson et al. 2021), etc. Usually, these tests are categorized as either convergent or divergent. A convergent task will usually require someone to think about a problem with a single solution, while a divergent task allows for multiple solutions. RAT and the Candle problem are examples of convergent tasks with a single solution; the Alternative Uses Task and DAT are divergent tasks, allowing the subject to come up with many solutions to a problem. All of these tests challenge an individual's ability to escape from a bias:

1. **Candle Problem**: The task involves providing a subject with a matchbook, a box of tacks, and a candle, with the objective being to attach the candle to a wall and light it without getting wax on the floor. The solution to this test is to use the box that holds tacks as a basin to hold the candle and collect the wax. The main focus of the candle problem is functional fixedness: the unlikeliness of the subject to use an object other than for its intended purpose. The box with tacks presents itself as a box meant for holding tacks and not for holding a candle and so a subject rarely uses the box for the test. Later experiments show that you can discourage functional fixedness based on how you present the information. For example, (Frank and Ramscar 2003) underlines the words "matchbook", "tacks", "candle" and "*box*" which doubled the number of subjects that solved the candle problem.
2. **Alternative Uses Task**: This task evaluates a subject's ability to name alternative uses of common objects, such as a brick, by using metrics such as the number of proposed uses and the uniqueness of each suggestion. Similar to the Candle Problem, the Alternative Uses Task highlights the challenge of overcoming functional fixedness biases by prompting participants to envision novel applications for familiar objects.
3. **Remote Associates Task**: A subject is provided a list of three remotely associated words and tasked with identifying a fourth word that links all three, e.g. fish, mine, rush $\rightarrow$ gold. Usually, the words are associated in different contexts: some might be parts of a compound word, others synonyms, and still others semantically similar. The biases in these experiments are examples of semantic and associative priming. The term "remote" underscores the departure from common word associations and therefore common human biases such as commonly associated words, e.g. dog, cat, horse $\rightarrow$ animal.
4. **Divergent Association Task**: DAT can be seen as a divergent counterpart to RAT; the subject must name n words that are as far from each other semantically as possible. This task challenges individuals to overcome biases inherent in word association, where the tendency to conceive of semantically similar words often inhibits one's ability to find semantically dissimilar words.

Creativity Versus Proficiency

Creativity tests are not proficiency tests. If a subject is tasked with solving a creative test like the candle problem but is provided the solution beforehand, they may "pass" the test by simply recalling the solution. Tests such as history or doctoral examinations often include memorization of factual information as a measure of success in the test. However, proficiency in reproducing a memorized solution does not equate to success in a creative test.

What if the test precludes memorized solutions? Consider an algebra problem like $nx - y = 0$ with a varying n and y so that memorized solutions are impractical. In such cases, proficiency is demonstrated through the application of memorized procedural steps. However, the ability to utilize a memorized procedure to solve a problem also does not signify passing a creative test.

The distinguishing factor between tests of creativity and tests of proficiency lies in the extent to which bias must be overcome. In instances where the answer or procedure is memorized, subjects do not confront bias. However, introducing bias that the subject must overcome makes the test a measure of creativity. Moreover, it stands to reason that the more difficulty the subject may have in overcoming that bias, the better the test for measuring creativity.

Contrasting Creative Artefact Generation and Creative Process

Under the premise that the creative process requires the ability to escape bias, one may be perceived as creative from a societal standpoint without being perceived as creative from a personal one. Any creative society (artists, dancers, mathematicians, etc.) may be viewed as a confluence of biases that results in a distinctive (amalgamated) bias. Consequently, an individual may be deemed creative when their work lies beyond the boundaries of the group's collective bias. This does not necessarily imply that the individual has engaged in the creative process; rather, it reflects that their output diverges from the collective bias. Even in instances where the group analyzes the individual's creative process, the uniqueness perceived by the group is predicated upon the individual's departure from the collective bias.

$8809 = 6$	$8603 = 4$
$7856 = 3$	$9762 = 2$
$4646 = 4$	$3315 = 0$
$1234 = 1$	$6666 = ?$

Figure 1: What is the solution to this problem?

For example, when given the problem shown in Figure 1, a mathematician's inclination is to view it through a numerical lens—due to their developed mathematical bias, they completely overlook the perspective that numbers can be perceived as shapes. Here, the solution is to sum the number of holes on the left side of the equation, i.e. $\{8\}$ has 2 holes, $\{0, 4, 6, 9\}$ have 1 hole, and $\{1, 2, 3, 5, 7\}$ have 0 holes, so 6666 has $1 + 1 + 1 + 1 = 4$ holes. If you ask a kindergartener to solve this problem, they are much more likely than the mathematician to see the pattern, because they have different biases than mathematicians. If, in a hypothetical scenario, a roomful of mathematicians witnesses a child successfully solving the problem, they may regard the child as creative. However, if subsequent children also independently solve the problem, the perception of the initial child's creativity may be reconsidered.

In *Surely You're Joking, Mr. Feynman* (1985) Feynman reflects on how others perceived him as a mathematical genius because he could solve problems that they couldn't. Feynman had a different perspective. He attributed his problem-solving not to any innate superiority, but rather to his diverse set of problem-solving techniques not commonly found among his peers. When others' conventional methods failed to solve the problem, Feynman was able to solve them with his unconventional methods that appeared brilliant in comparison. It is worth noting that this perspective does not diminish Feynman's exceptional abilities but rather sheds light on how he believed his unique problem-solving toolkit contributed to his reputation.

The book *Rookie Smarts* (Wiseman 2015) underscores the advantages rookies have within their workplace, over industry veterans, because they aren't shackled by preconceived notions of "how things are supposed to be done".

These examples emphasize the notion that an individual's being deemed creative from a group's perspective does not necessarily signify that the individual has in fact undertaken a creative process. However, the value of such artefacts created without active engagement in the creative process should not be diminished, as they can serve to challenge biases and foster creativity within the group.

The Time-Traveling Artist Let us consider a hypothetical scenario in which an artist memorizes and replicates renowned works of art. Afterward, the artist makes trips back in time to a period just before each great work of art was created by the original artist to present his own plagiarized work. In such a situation, society would characterize the artist as creative. However, given the context, the artist should not, in fact, be labeled creative but rather a plagiarist (Ventura 2016). While society may afford them the title of a creative artist, their endeavors do not authentically embody the creative process, again reemphasizing the point that society's evaluation of creativity does not imply that the product so judged was a result of engagement in the creative process.

Computational Creativity

Given the dichotomous possibilities of being judged creative while engaging in a creative process and while not engaging in a creative process, evaluation of CC systems may be effected within either paradigm. CC systems do not have to engage in a creative process to be useful or to output creative artefacts. However, if the aim is for a CC system to engage in the creative process, either to better understand the process itself or because it may enhance the creativity of its artefacts, then the system must overcome its algorithmic biases. The next section constructs a mathematical framework for characterizing bias in the context of computational creativity and for analyzing the ability of a CC system to overcome bias.

Mathematical Framework

Creative System Framework We will describe the CSF using Ritchie's simplified notation (Ritchie 2012), although we will make slight notational changes. As with Ritchie's notation, $tuples(\mathcal{X})$ denotes the set of finite tuples of set $\mathcal{X}$ and $elements()$ is a function that reverts a tuple into a set. The CSF is built upon the universe of concepts $\mathcal{U}$, which contains the set of all possible (in)complete concepts, e.g. $c \in \mathcal{U}$, including the empty concept $\top$. Three main functions are used to interact with $\mathcal{P} \subseteq \mathcal{U}$, namely the value function $V : \mathcal{P} \to [0, 1]$, acceptability function $N : \mathcal{P} \to [0, 1]$, and exploration function $Q : tuples(\mathcal{P}) \to tuples(\mathcal{P})$.[1] The output of V determines how valued a concept is, and the output of N determines level-of-membership for a concept in the domain measured by N (e.g. classic rock, music, movies, mathematics, etc.). The search function Q is responsible for finding new concepts given some inspiring sequence I of concepts.

An exploratory system can be described as a 4-tuple $(\mathcal{P}, V, N, Q)$, and its search over time can be described as a dynamical system:

$$I_{t+1} = Q(I_t)$$

[1] In Ritchie's notation Q is technically a mapping to mappings, i.e. $Q : [0, 1]^{\mathcal{P}} \times [0, 1]^{\mathcal{P}} \to (tuples(\mathcal{P}) \to tuples(\mathcal{P}))$, where $[0, 1]^{\mathcal{X}}$ denotes the set of all possible mappings of $\mathcal{X}$ to a real number between 0 and 1 inclusive, and notationally is represented as $Q(V, N)(I)$, but we elect to use a simpler notation and assume V and N are "baked into" Q.

where I_0 is an initial inspiring sequence of concepts, which might be empty or contain only the empty concept $\top$. Another function used to understand exploratory systems is the reachability function:

$$\rho(Q, h, I_0) = \bigcup_{t=0}^{h} elements(I_t)$$

which returns all reachable concepts for a given search strategy Q, time horizon h, and initial inspiring sequence I_0.

If $\mathcal{P}' \subset \mathcal{U}$ is a set containing (V, N, Q) triplets then a system $(\mathcal{P}', V', N', Q')$, where V', N', and Q' operate on $\mathcal{P}'$, is called a transformational system. We will make use of Wiggins' assumption that all value functions V in P' are static. We will continue to use $'$ to differentiate between exploratory systems $(\mathcal{P}, V, N, Q)$ and transformational systems $(\mathcal{P}', V', N', Q')$.

The search dynamics of a transformational system are usually described in the same way as an exploratory system's search dynamics; however, implicitly a transformational system could run and analyze the search performed by exploratory systems to determine their level of promise. Since it is generally uncomputable to determine how a program will function without running it, transformational search will likely involve running exploratory systems. We will make this capability explicit by defining the search dynamics of a transformational system as:

$$I'_{t+1}, \bar{I}_t = Q'(I'_t, I_{0:t}) \quad (1)$$

$$V, N_{t+1}, Q_{t+1} = \sigma(I'_{t+1}) \quad (2)$$

$$I_{t+1} = Q_{t+1}(\bar{I}_t) \quad (3)$$

where I_0 and I'_0 are initial inspiring sequences and I'_0 must include at least the empty concept $\top$.

At a high level, the dynamics involve a search step from the transformational system (step 1), the selection of an exploratory system with the select function $\sigma : tuples(\mathcal{U}) \to \mathcal{U}$ (step 2), and performing a search step from the selected exploratory system (step 3).

Note the change in the domain and range of Q' (when compared with Q). The domain has been expanded to incorporate not only its own inspiring sequence I'_t but also the sequence of all previous inspiring sequences given to past exploratory systems, $I_{0:t} = (I_0, I_1, \ldots, I_t)$. This domain change is also propagated to both V' and N', and $I_{0:t}$ serves as feedback from the exploratory systems to the transformational system, allowing the transformational system to make changing evaluations about exploratory systems as more information is received by running them. This feedback loop between the transformational and exploratory systems is key for understanding bias and the overcoming of bias for CC systems.

The range of Q' has also been modified (compared with Q) to include $\bar{I}_t$, a filtered version of the exploratory system's inspiring sequence that will be given to the (selected) exploratory system. There are multiple ways Q' can filter I_t, but we consider only two: stateless:

$$\bar{I}_t = \begin{cases} I_0 & \text{if } (V, N_t, Q_t) \neq (V, N_{t+1}, Q_{t+1}) \\ I_t & \text{otherwise} \end{cases}$$

and stateful:

$$\bar{I}_t = \begin{cases} \bigoplus_{k=0}^{t} I_k & \text{if } (V, N_t, Q_t) \neq (V, N_{t+1}, Q_{t+1}) \\ I_t & \text{otherwise} \end{cases}$$

where $\oplus$ is concatenation.

Another notable change (from the original dynamics) is the select function σ, which selects a single exploratory system[2] to run at timestep $t+1$. Here, we assume that σ retrieves the first element from a sequence and assume that all Q' are well-formed in the sense that they put the exploratory system intended for selection at the beginning of their updated inspiring sequence I'_{t+1}. The selected exploratory system can be an incomplete exploratory system or even the empty concept $\top$; we interpret all such systems' search functions as the identity function, i.e. $Q(I_t) = I_t$. To include the possibility that σ may function differently than described here, the definition of a transformational system is a 5-tuple $(\mathcal{P}', V', N', Q', \sigma)$ that includes σ.

Bias Let $V_\alpha(\mathcal{P}) = \{c \in \mathcal{P} \mid V(c) > \alpha\}$ be the set of all concepts with a value greater than threshold α. If $\rho(Q, h, I_0) \cap V_\alpha(\mathcal{P}) = \emptyset$ then the system is described as being uninspired (Wiggins 2006). Wiggins provides different classifications of uninspiration, but here we focus specifically on generative uninspiration, where the system can't find highly-valued concepts due to some limitation of its search capability—an issue that can only be resolved by updating Q and/or N.[3] We will refer to such a system as uninspired, but may also say that Q is an uninspired algorithm or search strategy. If $\rho(Q, h, I_0) \cap V_\alpha(\mathcal{P}) \neq \emptyset$ then we characterize the system as inspired and may say that Q is an inspired algorithm or search strategy. It should be noted that characterizing a search algorithm Q as (un)inspired is dependent on h, I_0 and $V_\alpha(\mathcal{P})$; for example, an exploratory system that is uninspired with respect to $V_\alpha(\mathcal{P})$ given h and I_0 may become inspired with respect to $V_\alpha(\mathcal{P})$ given more time or different initial conditions.

[2]While a transformational system can run Q' multiple times before running a new Q, run Q multiple times before running Q', or run multiple Q in parallel, we lose no generality describing transformational systems with our search dynamics. Running Q' multiple times before running Q is accomplished by having σ select $\top$, whose search operates as the identity function, allowing Q' to search for new Q without adding any additional concepts. Running Q multiple times before running Q' is accomplished by Q' not updating I'_t and σ continually selecting $Q_{t+1} = Q_t$. Running multiple Q in parallel can always be described by a single search in the same way that any multi-tape Turing machine can be reduced to a single-tape Turing machine (Sipser 2013).

[3]Recall that V and N are "baked into" in Q, so N influences Q's search.

Given the definitions of inspired and uninspired algorithms, we can define what it means for a system to be biased and to overcome bias. Let

$$v'_{u,t} = \max_{\substack{(V,N,Q)_k \in I'_t \\ \wedge Q \text{ uninspired}}} \{V'((V,N,Q)_k, I_{0:t-1})\}$$

be the value of the V'-maximized *uninspired* exploratory system at time t (or zero if there are no uninspired exploratory systems). And similarly let

$$v'_{i,t} = \max_{\substack{(V,N,Q)_k \in I'_t \\ \wedge Q \text{ inspired}}} \{V'((V,N,Q)_k, I_{0:t-1})\}$$

be the value of the V'-maximized *inspired* exploratory system at time t (or zero if there are no inspired exploratory systems). A system overcomes β-bias over h timesteps if $\exists t_0, t$ where $0 < t_0 < t \leq h$ and the following conditions hold:

Condition 1: $v'_{u,t_0} - v'_{i,t_0} > \beta$

Condition 2: $v'_{i,t} - v'_{u,t} > \beta$

Condition 3: $elements(I_t) \cap V_\alpha(\mathcal{P}) \neq \emptyset$

Condition 1 means that initially the best uninspired strategy is valued significantly more than any inspired one. Condition 2 means that eventually the situation is reversed and the best inspired strategy is valued significantly more than any uninspired one. Finally, Condition 3 states that at timestep t (the same as Condition 2), an inspired Q_t was selected, and it found a highly valued concept. **A transformational system is biased** when the first condition is true. **A transformational system has overcome bias** when all three conditions are true, i.e. the system is biased (Condition 1) and it overcomes bias (Conditions 2 and 3).[4]

The threshold β is included in Condition 1 for two reasons: the system may require a burn-in period—a length of time during which Q' can search for exploratory systems and establish its "beliefs" about the exploratory systems; the system may initially arbitrarily select uninspired search strategies or may do so purposefully to gain useful feedback. Only once the system is confident that an uninspired strategy is better than any inspired strategy, by a margin of β, is the system considered biased.

The threshold β is included in Condition 2 in order to disqualify situations in which the system "gets lucky" by stumbling on an inspired strategy that yields a high-value concept without being confident that that strategy is more valuable than the original uninspired one—overcoming bias should include both discovering new useful approaches *and* abandoning old ones that do not work. Finally, as conclusive proof that the transformational system has, in fact, overcome bias, it must discover a highly-valued concept (Condition 3).

Determining if a system has overcome bias only makes sense if the system is rational, since a system can purposefully choose poor search strategies to which it arbitrarily assigns high values initially before later lowering the value and choosing a known search strategy that finds high-valued concepts. Finding a formal definition for rationality is left to future work, and we will only informally define a rational system as one that makes decisions to maximize V. Therefore, a system that purposefully chooses poor search strategies would be considered irrational.

Feedback The definition of overcoming bias implies that if $\beta \geq 0.5$ then V' *must* utilize the feedback $I_{0:t}$ in order for the system to overcome bias, i.e. if $\forall i, j\ V'((V,N,Q), I_{0:i}) = V'((V,N,Q), I_{0:j})$ then Conditions 1 and 2 will never hold.[5] Furthermore, if Q' is trying to optimize V' then, at least indirectly, it must also utilize feedback.

An interesting aspect of the feedback mechanism employed by Q' is that for stateful $\bar{I}_t$, the search strategies Q can affect each other. Let Q_u and Q_i be uninspired and inspired search strategies, respectively, given h, I_0, and $V_\alpha(\mathcal{P})$. If an arbitrary Q runs for the first $t < h$ steps followed by Q_i running the remaining steps, then it is possible that Q_i might no longer be inspired with respect to $V_\alpha(\mathcal{P})$ because of $\bar{I}_t$. The opposite can also occur—some Q could run first followed by Q_u, but since $\bar{I}_t$ is stateful, then Q_u could become inspired with respect to $V_\alpha(\mathcal{P})$.

Pseudo-Transformational Systems

The definitions of bias and overcoming bias require a transformational system because the definitions rely on a comparison between different exploratory systems evaluated and selected by a transformational system. Therefore it is important to distinguish between transformational systems and exploratory systems. While it may seem that the two can be distinguished by their definitions alone, the illustrative example below shows why some systems that appear superficially to be transformational systems are, in fact, operationally equivalent to an exploratory system **in terms of the exploratory-level concepts they generate**; such systems can therefore be reduced to exploratory systems and should be considered only *pseudo-transformational*.

As an illustration of this idea, consider a system A

[4]Because the definition of overcoming bias involves h, it is necessary for the transformational functions Q' and V' (and possibly N') to have access to h in order to provide timely search and evaluation. For the sake of parsimony, we will assume that h is baked into the transformational functions.

[5]If $\beta \geq 0.5$, then in order for Condition 1 to hold, it must be the case that $v'_{u,t_0} > 0.5$. It follows that if V' doesn't change with feedback, $v'_{u,t}$ is bounded below by 0.5 for all time steps $t \geq t_0$. This means for Condition 2 to hold, $v'_{i,t}$ must be greater than $v'_{u,t_0} + 0.5$ which implies $v'_{i,t} > 0.5 + 0.5$ which implies $v'_{i,t} > 1$, but since the range of V' is $[0, 1]$, Condition 2 cannot hold.

Algorithm 1 System A is a pseudo-transformational system.

1: **procedure** $Q'(I'_t, I_{0:t})$
2: $(V, Q^*) = \sigma(I'_t)$
3: **for** i in $1, \ldots, n$
4: $\theta_i \sim \mathcal{N}$
5: $I_{t+1} = Q_{\theta_i}(I_t)$
6: **if** $V(\sigma(I_{t+1})) > V(\sigma(I_t))$
7: $Q^* = Q_{\theta_i}$
8: **return** $(Q^*), I_t$
9:
10: **procedure** $Q_\theta(I_t)$
11: $c^* = \sigma(I_t)$
12: **for** j in $1, \ldots, m$
13: $z_j \sim \mathcal{N}$
14: $c = G(z_j; \theta)$
15: **if** $V(c) > V(c^*)$
16: $c^* = c$
17: **return** (c^*)

Algorithm 2 System B is an exploratory system that is equivalent to system A.

1: **procedure** $Q(I_t)$
2: $c^* = \sigma(I_t)$
3: **for** i in $1, \ldots, n$
4: $\theta_i \sim \mathcal{N}$
5: **for** j in $1, \ldots, m$
6: $z_j \sim \mathcal{N}$
7: $c = G(z_j; \theta_i)$
8: **if** $V(c) > V(c^*)$
9: $c^* = c$
10: **return** (c^*)

(refer to Algorithm 1) containing Q_θ, V, Q', and V'.[6] This system will be composed of a generative neural network G, parameterized by θ, that takes an input vector z (sampled from a Gaussian $\mathcal{N}$) and converts it into a concept. If the parameters θ are interpreted as part of the program defining Q, then a change in the parameters θ is a change of the program Q. $Q_\theta(I_t)$ (line 10) is then defined as a function that generates m new concepts by randomly sampling m different z vectors; generating a concept c with $G(z; \theta)$ for each one; and returning the V-max concept among the m new concepts and the concepts found in I_t.

The parameters θ can be optimized to create better concepts according to some objective. This means that the optimization process can be interpreted as Q' and the optimization objective as V'. Let Q' (line 1) randomly create n different Q_θ by randomly sampling n parameter vectors and return the V'-max Q_θ among the newly generated Q_θ and the previously found Q_θ in I'_t, where

$$V'(Q_\theta, I_{0:t}) = V(\sigma(Q_\theta(I_t)))$$

and σ selects the first and only (in this example) concept returned by $Q_\theta(I_t)$.[7] Note that V is called directly in place of V' (line 6).

An exploratory system B that is equivalent to A may be constructed (refer to Algorithm 2). Instead of Q randomly sampling z vectors, let it randomly sample both z and θ and return the V-maximized concept. Because V' in system A does nothing more than act as a wrapper for V, the exploratory-level concepts generated by both systems will be identical and therefore they are fundamentally the same system, demonstrating that system A is only pseudo-transformational.[8]

In contrast, if system A was instead constructed such that its V' incorporated information beyond that provided by V—e.g., a measure of the complexity of Q, the diversity of the generated concepts, aesthetic information, etc.—the system would not have been reducible to an equivalent exploratory system and therefore, would be a true transformational system.

Designing a Creativity Benchmark

(Spendlove and Ventura 2022; Spendlove and Brown 2023) argue for exploring creative domains with a well-defined and objective V. Both studies suggest the domain of creative games, which can be formulated as Markov games (Littman 1994), as a candidate for further exploration. We propose a simplification of such environments to a single-player Markov game, i.e. a Markov decision process (MDP) (Puterman 1994) as the basis for testing the ability to overcome bias. An MDP is a 5-tuple $(\mathcal{S}, \mathcal{A}, T, R, \pi_0)$, where $\mathcal{S}$ is a set of states, $\mathcal{A}$ a set of actions, T a transition function, R a reward function, and π_0 the start state distribution. The transition function $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ defines the probability of transitioning from one state to another given an action. The reward function $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ assigns a real-value reward given a state s_t, action a_t, and next state s_{t+1}. If R is constrained to output a reward in the range $[0, 1]$, it can be used as V for an

[6] The rest of the parameters, e.g. N, are unnecessary for this argument.

[7] While we are violating our definition of a transformational system's search dynamics by having Q' and V' run Q and V internally, we do so to keep the example as simple as possible. Strictly following the definition would involve Q' selecting new Q and putting them all in $I'_1 = (Q_{\theta_1}, \ldots, Q_{\theta_n})$; cycling through the order, i.e. $I'_2 = (Q_{\theta_2}, Q_{\theta_3}, \ldots, Q_{\theta_n}, Q_{\theta_1})$ $\ldots I'_n = (Q_{\theta_n}, Q_{\theta_1}, \ldots, Q_{\theta_{n-2}}, Q_{\theta_{n-1}})$; and finally choosing Q^* by evaluating $I_{0:t}$. The point being demonstrated here is that this additional complexity is important for providing the feedback necessary for true transformational creativity but it is misused or ignored in pseudo-transformational systems.

[8] This equivalence argument only applies to rational systems, as it may be possible to construct an irrational exploratory system that mimics a transformational system.

Algorithm 3 Given an MDP $M = (\mathcal{S}, \mathcal{A}, T, R, \pi_0)$, transformational system $X = (\mathcal{P}', V', N', Q', \sigma)$, horizon h, bias threshold β, and value threshold α.

1: **procedure** HasOvercomeBias(M, X, h, β, α)
2: $s_0 \sim \pi_0$
3: $I_0 = \{(\cdot, \cdot, s_0)\}$ ▷ '·' is a dummy variable
4: $I'_0 = \{\top\}$
5: isBiased = False
6: **for** t in $1, \cdots, h$
7: $I'_t, I_{0:t} =$ SystemSearch$(X, I'_{t-1}, I_{0:t-1}, T)$
8: $m =$ Margin$(M, X, I'_t, I_{0:t-1}, h, \alpha)$
9: **if** $m < -\beta$
10: isBiased ← True
11: $s_{t-1}, a_{t-1}, s_t = \sigma(I_t)$
12: **if** $R(s_{t-1}, a_{t-1}, s_t) > \alpha$
13: **if** $m > \beta$
14: **if** isBiased
15: **return** True
16: **return** False
17: **return** False
18:
19: **procedure** SystemSearch($X, I'_t, I_{0:t}, T$)
20: $I'_{t+1}, \bar{I}_t = Q'(I'_t, I_{0:t})$
21: $V, N_{t+1}, Q_{t+1} = \sigma(I'_{t+1})$
22: $I_{t+1} = Q_{t+1}(\bar{I}_t; T)$
23: **return** $I'_{t+1}, I_{0:t+1}$
24:
25: **procedure** Margin($M, X, I'_t, I_{0:t-1}, h, \alpha$)
26: iVals = {} ▷ inspired systems' values
27: uVals = {} ▷ uninspired systems' values
28: **for** $(V, N, Q) \in I'_t$
29: **if** Inspired$(M, (V, N, Q), I_{t-1}, h, \alpha)$
30: iVals = iVals $\cup \{V'((V, N, Q), I_{0:t-1})\}$
31: **else**
32: uVals = uVals $\cup \{V'((V, N, Q), I_{0:t-1})\}$
33: **return** max iVals − max uVals
34:
35: **procedure** Inspired($M, (V, N, Q), I_t, h, \alpha$)
36: **for** k in $0, \cdots, h-1$
37: $I_{t+k+1} = Q(I_{t+k}; T)$
38: $s_k, a_k, s_{k+1} = \sigma(I_{t+k+1})$
39: **if** $R(s_k, a_k, s_{k+1}) > \alpha$
40: **return** True
41: **return** False
42:
43: **procedure** $Q(I_t; T)$
44: $s_{t-1}, a_{t-1}, s_t = \sigma(I_t)$
45: $a_t = Q_a(s_t)$
46: $s_{t+1} \sim T(s_t, a_t)$
47: $I_{t+1} = (s_t, a_t, s_{t+1}) \oplus I_t$
48: **return** I_{t+1}

exploratory system.[9] π_0 is a distribution over $\mathcal{S}$ from which the initial state s_0 is sampled.

Using the MDP framework makes testing whether a system has overcome bias relatively easy. Each state can be treated as a concept by letting $\mathcal{P} = \mathcal{S}$ and therefore transitions are rewarded according to the value of the next state: $R(s_t, a_t, s_{t+1}) = V(s_{t+1})$. While technically Q should output a sequence of states/concepts directly, because MDPs require actions, some part of Q should output actions. A rigorous treatment of the (related) idea of incorporating MDPs into the CSF is provided by the creative action selection framework (Linkola, Guckelsberger, and Kantosalo 2020).

We provide here a simplified version, leaving the finer details for future work, and show how a creative task formulated as an MDP can be utilized to test whether a system $(\mathcal{P}', V', N', Q', \sigma)$ has overcome bias (shown in Algorithm 3). After initializing variables (lines 2-5), the search for exploratory systems is performed out to the horizon h (line 6). At each timestep, a single system search is performed (line 7), using the search dynamics of a transformational system described above (lines 19-23). For convenience, an exploratory system Q (line 43) is composed of an action selection algorithm Q_a (line 45) and the transition function T (line 46) and maps state-action pairs to next states. Q keeps a history of the transitions as part of I (line 47). After stepping through the system search and the MDP, the margin between the highest-valued *inspired* search strategy in I'_t and the highest-valued *uninspired* search strategy in I'_t (lines 25-33) is computed (line 8) to determine if the former is valued significantly *less* than the latter (lines 9-10). A search strategy is categorized as inspired or uninspired depending on whether or not it finds a valuable concept within the horizon h (lines 35-41). Next, the (state, action, next-state) transition selected by Q (line 11) is tested to determine if the system has discovered a highly-valued state (line 12). If it has, the system's highest-valued inspired search strategy is again compared against its highest-valued uninspired search strategy to see if the former is now valued significantly *more* than the latter (line 13). If both statements are true, satisfying Conditions 2 and 3 respectively, and if the system was previously biased (line 14), satisfying Condition 1, then `TRUE` is returned (line 15) because the system has overcome bias. If all the conditions are not satisfied then `FALSE` is returned (lines 16-17) because it has not.

A useful aspect of this formulation is that toy problems can be designed for CC systems with relatively small state spaces, allowing for quick testing and iteration. Therefore, any classical MDP toy problems used in reinforcement learning (RL), such as Frozen Lake,[10]

[9] Because MDPs rely on the Markov property, the reward function cannot represent all V, but the space of MDPs is rich and complex enough that this is not a major limitation.

[10] https://gymnasium.farama.org/environments/toy_text/frozen_lake/

can be employed for testing CC systems in the manner shown here. This begs the question: if a creative problem is modeled as an MDP, and RL algorithms are designed to optimize MDPs, does that mean RL systems engage in a creative process? We argue that in general the answer is no, because many RL systems simply maximize average reward, which implies that V' is directly maximizing V, and therefore these systems should be considered only pseudo-transformational, per the argument made in the section above. However, in some cases, RL systems may include intrinsic rewards or leverage other active learning paradigms that utilize intrinsic values to improve the exploration of a system. Such a system would not be simply directly optimizing V with V', and, therefore, this subset of RL systems might be looked at as a mechanism, or at least an inspiration, for designing CC algorithms that can overcome bias.

While we have asserted here that overcoming bias is a necessary characteristic of the creative process, we are not ready to claim that it is a sufficient one. As a result, while any MDP may be sufficient for testing if a system has overcome bias, it may not be a strong enough test to support a claim of creativity. More work is required to characterize MDPs that may be sufficient for a test of creativity.

While toy problems are useful, it is more interesting and applicable to design larger, more complex MDPs with sparse and even deceptive reward functions that will serve as better proxies for real-world scenarios requiring creativity. For example, by simulating the physics of versatile components such as Lego blocks, ramps, pulleys, hammers, dominoes, etc., and tasking a system with the objective of transporting a ball from an initial position to a goal position, the systems' ability to design Rube Goldberg machines can be evaluated. The physics of the environment constrain the conceptual space, giving us an objective N, and checking whether the ball is in the goal position serves as an objective V. Markov games, such as Codenames (Spendlove and Ventura 2022), with static agents are also MDPs and can serve as complex, but objective, benchmark tests.

Because the definition of overcoming bias is based on a horizon h, it is important to discuss how time is measured in this evaluation. We suggest three options with their pros and cons:

1. *Number of Q applications*: This can alternatively be thought of as the number of steps the system interacts with the environment. This makes sense because we have defined the search process of creative systems as a dynamical system in which each step involves an application of Q. Unfortunately, the application of Q may not be constrained enough for this to always be a reasonable approach. For example, if the system creates a model of the environment and then simulates runs within that model, then Q can potentially run for an arbitrary number of virtual steps before making a single step in the real environment.
2. *Wall clock time*: This is an easy method to use, but it might unfairly benefit multi-processor systems that can run multiple Q strategies in parallel or single search strategies that utilize parallel processing for faster "search" (such as generative neural networks).
3. *Computation steps*: This is perhaps the most fair method. However, this might benefit bespoke, special-purpose systems over more general-purpose systems. Using the generative neural network as an example, it takes billions of computations to generate a single concept, while a special-purpose system might take only a thousand. On the other hand, there may be an interesting trade off between the speed of convergence of special-purpose systems and the ability of general-purpose systems to more effectively overcome biases.

Conclusion

Determining whether a system has overcome algorithmic bias is fundamental to determining if the system has engaged in the creative process. In this work, we have outlined definitions for inspired and uninspired algorithms, transformational search dynamics, biased systems, and systems that overcome bias. While our definitions provide necessary conditions to characterize a system as being biased or overcoming bias, further constraints can be added to strengthen the conditions, such as enforcing $v'_{u,t_0} - v'_{i,t_0} > \beta$ for multiple timesteps.

We have also suggested a simple methodology for testing a system's ability to overcome bias by utilizing MDPs as a framework. MDPs afford us considerable flexibility in designing creative tests that span simple toy problems to complex creative domains. While our methodology is a pass/fail evaluation, averaging pass/fail outcomes across multiple tests can yield a more nuanced assessment. It should be noted, though, that *failing a creative test does not mean that the system cannot in general overcome bias* because a test is specific to β, h, and $V_\alpha(\mathcal{P})$.

Overcoming bias, in combination with criteria proposed in (Spendlove and Brown 2023; Spendlove and Ventura 2022), moves us one step closer to designing suitable benchmarks for CC systems. While it is plausible that algorithms excelling on such benchmarks may still exhibit domain-specific biases—overcoming biases detectable by specific creative tests while struggling with biases in other creative domains—we remain optimistic that a (set of) useful CC benchmark(s) can be developed and that doing so will catalyze innovation and progress in the field of CC.

References

[Boden 1991] Boden, M. A. 1991. *The Creative Mind: Myths and Mechanisms*. London and New York: Routledge.

[Duncker 1974] Duncker, K. 1974. *Zur Psychologie des Produktiven Denkens*. Springer Berlin Heidelberg.

[Feynman 1985] Feynman, R. P. 1985. *"Surely You're Joking, Mr. Feynman!": Adventures of a Curious Character*. New York and London: W. W. Norton Company.

[Frank and Ramscar 2003] Frank, M. C., and Ramscar, M. 2003. How do presentation and context influence representation for functional fixedness tasks? In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 25, 1345.

[Guilford 1967] Guilford, J. P. 1967. Creativity: Yesterday, today and tomorrow. *The Journal of Creative Behavior* 1(1):3–14.

[Jordanous 2022] Jordanous, A. 2022. Should we pursue SOTA in computational creativity? In *Proceedings of the 13th International Conference on Computational Creativity*, 159–163. Association for Computational Creativity.

[Linkola, Guckelsberger, and Kantosalo 2020] Linkola, S.; Guckelsberger, C.; and Kantosalo, A. 2020. Action selection in the creative systems framework. In *Proceedings of the 11th International Conference on Computational Creativity*, 303–310. Association for Computational Creativity.

[Littman 1994] Littman, M. L. 1994. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the 11th International Conference on Machine Learning*, 157–163.

[Loughran 2022] Loughran, R. 2022. Bias and creativity. In *Proceedings of the 13th International Conference on Computational Creativity*, 354–358. Association for Computational Creativity.

[Mednick 1962] Mednick, S. A. 1962. The associative basis of the creative process. *Psychological Review* 69:220–232.

[Olson et al. 2021] Olson, J. A.; Nahas, J.; Chmoulevitch, D.; Cropper, S. J.; and Webb, M. E. 2021. Naming unrelated words predicts creativity. *Proceedings of the National Academy of Sciences* 118(25):e2022340118.

[Puterman 1994] Puterman, M. L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. USA: John Wiley & Sons, Inc.

[Ritchie 2012] Ritchie, G. 2012. A closer look at creativity as search. In *Proceedings of the Third International Conference on Computational Creativity*, 41–48. Open University Press.

[Sipser 2013] Sipser, M. 2013. *Introduction to the Theory of Computation*. Boston, MA: Cengage Learning, third edition.

[Spendlove and Brown 2023] Spendlove, B., and Brown, D. 2023. What makes gameplay creative? In *Proceedings of the 14th International Conference on Computational Creativity*, 98–101. Association for Computational Creativity.

[Spendlove and Ventura 2022] Spendlove, B., and Ventura, D. 2022. Competitive language games as creative tasks with well-defined goals. In *Proceedings of the 13th International Conference on Computational Creativity*, 291–299. Association for Computational Creativity.

[Ventura 2016] Ventura, D. 2016. Mere generation: Essential barometer or dated concept? In *Proceedings of the Seventh International Conference on Computational Creativity*, 17–24. Association for Computational Creativity.

[Wiggins 2006] Wiggins, G. A. 2006. A preliminary framework for description, analysis and comparison of creative systems. *Knowledge Based Systems* 19:449–458.

[Wiseman 2015] Wiseman, L. 2015. *Rookie Smarts: Why Learning Beats Knowing in the New Game of Work*. New York: Harper Collins.