On Collaborator Selection in Creative Agent Societies: An Evolutionary Art Case Study

Simo Linkola and Otto Hantula

Department of Computer Science, University of Helsinki, Helsinki, Finland
{simo.linkola, otto.hantula}@helsinki.fi

Abstract. We study how artistically creative agents may learn to select favorable collaboration partners. We consider a society of creative agents with varying skills and aesthetic preferences able to interact with each other by exchanging artifacts or through collaboration. The agents exhibit interaction awareness by modeling their peers and make decisions about collaboration based on the learned peer models. To test the peer models, we devise an experimental collaboration process for evolutionary art, where two agents create an artifact by evolving the same artifact set in turns. In an empirical evaluation, we focus on how effective peer models are in selecting collaboration partners and compare the results to a baseline where agents select collaboration partners randomly. We observe that peer models guide the agents to more beneficial collaborations.

Keywords: computational social creativity, evolutionary art, collaboration, learning from experience

1 Introduction

Finding good collaboration partners within a society is a multifaceted problem for creative agents. Artistic *skills* and *aesthetic preferences* over artifacts can be diverse across the agents, but they have a direct effect on the collaboration. We propose different peer modeling schemes to select collaboration partners for image production. With diverse agent societies we empirically determine how effective learned peer models are using an experimental collaboration process. The experiments show that peer modeling improves the collaboration outcomes.

This work draws from three fields. At large, it is situated in the context of *computational social creativity* (see, e.g. Saunders and Bown [1]); we consider a society of creative agents interacting with each other and focus on emergent properties arising from interactions. For creativity, our agents use techniques from the field of *evolutionary art* [2–4] to generate images, and their collaboration also takes advantage of properties of evolutionary methods. Lastly, we also aim to contribute to *self-adaptivity* [5] of agents, in particular, adaptivity of interaction with others: our agents utilize machine learning to gain interaction awareness, i.e. a capability to reflect on and control one's interaction [6].

In (computational) creativity, collaboration is seen as invaluable social behavior as it can lead to artifacts which might not be created by any single agent

alone [7–9]. Creative artifacts can be evaluated by different characteristics, e.g. value, novelty [10] and intentionality [11]. Our individual agents evaluate artifacts using computational aesthetics for *value* and use a memory model to compute their *novelty*. Our main contributions involve how an agent can learn to be *intentional* in its collaboration partner selection.

We have chosen evolutionary art as an example domain as it has been extensively studied before, and both skill modeling (using varying primitive sets) and aesthetic preferences (using computational aesthetic measures) are readily available. We do not claim that our single agent evolutionary art configuration is novel, but it serves as a basis for our collaboration experiments.

Although an agent has to model its peers' "minds" for it to distinguish favorable collaboration partners [12], there is a prominent lack of research directed towards collaboration partner selection in creative societies. Surprisingly, given how pivotal collaboration is in creativity, we are not aware of any previously formulated collaboration process models for independent agents creating evolutionary art with genetic programming.

In our experiments, we focus on a society of creative agents able to interact with each other by exchanging evaluations of created artifacts and through collaboration. Our goal is to add to the understanding of how peer models are able to capture inter-agent differences, i.e. aesthetic preferences and skills, and how different ways of learning peer models deviate from each other. For example, how does an *altruistic approach*, where an agent models who likes the artifacts it creates, differ from *selfish approaches*, where an agent models either the quality of collaboration with its peers or appreciation it has for its peers' artifacts.

The rest of the paper is organized as follows. After a brief overview of related work, and an overview of our agent society as a whole and individual agent behavior, we move to our contributions. We devise a collaboration process for agents creating evolutionary art. Further, we describe several possible schemes for peer model learning. We then give an experimental setup designed to evaluate both the collaboration process and the validity of the peer models learned with each scheme. Our main contributions are in Sect. 7, where we report our findings from our experiments. We end the paper with discussion and conclusions.

2   Related Work

We briefly cover related work on the fields of computational creativity and self-adaptivity, and evolutionary computation and art.

Computational creativity and self-adaptivity Computational social creativity [1] studies creative agents interacting with each other and is influenced by the systems view of creativity [13]. The collaboration process we propose is a form of alternating co-creation [14], and our agents are related to curious design agents [15] in how they use memory to compute novelty of observed artifacts.

Musical collaboration of computational agents has been studied in various contexts, such as with musebots [16] and in memetic melody generation and

evolution [17]. In storytelling, Pérez y Pérez et al. [9] have studied how creative systems with different knowledge bases may collaborate on story generation producing stories which combine the agents' knowledges.

Our agents model their interactions with their peers and use the learned models to guide their actions. Lewis et al. [18] call this type of self-adaptive behavior interaction awareness. In the context of computational creativity, self-adaptivity is called metacreativity, an ability to be aware and in control of one's own creative process [6]. Metacreativity is an essential concept in creative systems enabling the system to make informed decisions about its own operation, possibly leading to transformational creativity [10].

Creative systems modeling their interactions include, e.g. use of reinforcement learning for curious agents to induce new creative behaviors in agents [19], and user modeling in a recommender system where the model is used to predict which recommendations would inspire p-creative behavior in the user [20].

Our agents utilize machine learning to acquire awareness and control over their interactions. To this end, they relate to autonomous agents operating in dynamic environments, where reinforcement learning methods [21] are often used to guide an agent's behavior. This line of research involves, for example, a study of cooperative multi-agent systems using Q-learning [22] and, a general method for reasoning about the behavior and related parameters of other agents [23].

Evolutionary computation and art Creating visual artifacts using evolutionary computation has a long tradition (see, e.g. Romero and Machado (eds.) [3]). Sims [2] created the first images using genetic programming and Machado and Cardoso [24] studied early possibilities of computational aesthetics for images. More recently, Vinhas et al. [25] explored opportunities to include both fitness and novelty into a single evolutionary art system.

Our work draws from den Heijer and Eiben [4]. They cross-evaluate a set of computational aesthetic measures for evolutionary art. We use a subset of aesthetic measures present in their work with our independent creative agents and contrast some of our results to theirs.

Romero et al. [26] proposed a hybrid society model for evolutionary art consisting of humans, and computational creators and critics. In their model, the interaction between various entities is orchestrated through a central element. We study direct interaction between agents which are both creators and critics.

Lastly, our agent society and collaboration model can be seen to have relations to multi-objective evolutionary computation (see, e.g. Zhou et al. [27]) and island models [28].

3   Agent Society and Individual Agents

At large, we consider an agent society consisting of a diverse set of artistic agents seeking to generate creative, i.e. novel and valuable images. We are especially interested in how agents find favorable collaboration partners. Since agents differ

Table 1. The complete set of primitives (functions and terminals) used in our experiments. An individual agent knows all the terminals and basic math functions, but only a random sample ($n = 8$) of the other functions.

Terminals	x, y, ephemeral double $\in [-1, 1]$, π, golden ratio
Basic Math	plus, minus, divide, multiply, mod
Relational	min, max
Various Math	ln, log10, log2, sin, cos, sinh, cosh, tanh, atan, hypot, abs, sqrt abs, parabola, avg sum, sign, abs dist
Noise & Various	perlin (2D), perlin (1D), simplex (2D), plasma

in their image creating skills and aesthetic preferences, it seems beneficial that agents model their peers in order to differentiate their collaboration potential.

The agents have limited interaction capabilities: they can only evaluate each others' artifacts or generate an image in pairwise collaboration. The problem for each agent now is how to model their peers using the information from such interactions alone.

The society is run in an iterative simulation, where at uneven time steps agents create artifacts individually (we call these *solitary artifacts*) and at even time steps they collaborate. On each solitary time step an agent creates a new artifact, drawing inspiration from its own previous artifacts. On collaboration time steps agents create artifacts in pairs, drawing inspiration from both collaborators' previous artifacts.

After agents have created all the artifacts on a particular time step, they send the artifacts to their peers for feedback. Depending on the learning scheme, agents then utilize machine learning using the feedback to learn peer models. Agents also memorize artifacts they receive from their peers if they individually consider them good enough.

An agent creates images using evolutionary programming and evaluates them using computational aesthetics for value and memory for novelty. In this sense, an individual agent is related to evolutionary art methods utilizing both fitness and novelty in their artifact evaluation (see, e.g. Vinhas et al. (2016)) as well as to curiosity based agents [15]. Distinguishable in the proposed agent model is how an agent taps into its memory to initialize the population on consecutive evolutionary engine's executions. This allows an agent to continue working on earlier artifacts, potentially developing itself a niche in the creative space.

Next, we describe configurations of different components of agents.

Evolutionary engine An agent's evolutionary engine uses genetic programming to evolve *expression trees* [2] consisting of terminals (leafs) and functions (inner nodes), together called primitives. An expression tree accepts two arguments x and y. An image is produced by evaluating the function tree for every (x, y) pair in the image to define the value of each pixel. The created images are then evaluated using a fitness function.

The terminal and function types available to an agent are shown in Table 1. Each agent knows all terminals and basic math functions. In our experiments,

Table 2. Evolutionary engine's parameters for a single agent's solitary images.

Parameter	Value
Representation	Expression tree (max depth 8), see Table 1.
Initialization	Up to half of the population mutated from the agent's memory, the rest with ramped half-and-half (depth $\in [2, 6]$)
Selection	Double tournament (size 3), Elitist (best 1)
Mutation	Subtree mutation; rate $= 0.25$
Crossover	Subtree crossover; rate $= 0.75$
Fitness Function	Utilizes memory and aesthetics, see text.
Population size	20
Generations	10
Image properties	64x64, 8-bit greyscale

to simulate different skills of agents, each of the agents knows only a subset of all other functions. For each agent, we sample 8 functions using uniform distribution at agent's initialization time, i.e. an agent's skills do not change during its lifetime.

Largely, our agents have typical parameter settings for their evolutionary art engine (see Table 2). The most notable difference is the restricted population size as the execution time of the whole agent society is heavily affected by it.

Aesthetic measures The aesthetic measures in our experiments are a subset of aesthetics present in den Heijer and Eiben [4]: Benford's law (BLW), entropy (ENT), fractal dimension (FRD), global contrast factor (GCF) and symmetry (SYM) (representative images are seen in Fig. 1). Each individual agent is randomly assigned one aesthetic measure, which it uses to compute the value of artifacts it generates or observes. The aesthetic does not change during the simulation. We have selected this set of computational aesthetics as it has been studied before, and provides reasonably divergent measures to investigate their connections in collaboration.

Benford's law and entropy both measure distributional features from images. Fractal dimension and global contrast factor measure structural properties from image's layout. Symmetry combines image's observed vertical, horizontal and diagonal symmetries. The detailed description of each aesthetic measure is out of scope for this work, thus we guide the reader to den Heijer and Eiben [4] for exact formulations.

Memory model Each agent has a limited memory ($n = 500$) where it can store artifacts it observes. The memory is used to evaluate each observed artifact's *novelty*, and to initialize a population for each evolutionary engine's execution. If a new artifact is memorized when the memory is full, the oldest artifact is erased. The novelty with respect to an agent's current memory is computed similarly to Vinhas et al. [25], i.e. novelty$(I) = \text{ed}(I, m)$, where I is the artifact in question, m is the most similar artifact in the agent's memory, and $\text{ed}(\cdot)$ is the normalized Euclidean distance between the created greyscale images.

Fig. 1. Typical images generated by agents with different aesthetic measures. From left: Benford's law, entropy, fractal dimension, global contrast factor and symmetry.

Fitness An individual agent computes the fitness of an image using its aesthetic measure for value, and its memory for novelty, using the following formula:

$$\text{fitness}(I) = \begin{cases} \frac{1}{2}\text{value}(I) + \frac{1}{2}\text{novelty}(I), & \text{if PNG compress ratio} \geq 8\% \\ 0, & \text{otherwise} \end{cases}$$

where I is the artifact, value$(\cdot)$ is the aesthetic measure function, novelty$(\cdot)$ is computed using the memory as described above, and PNG compress ratio is the relative file size of PNG image over original image. The PNG compress ratio is used to filter out too simple images as uninteresting.

Initialization and evolution of a population Each time the evolutionary engine is executed, an agent first creates a new initial population consisting only of artifacts produced with the skills (primitives) the agent itself possesses. The population is partly created using the agent's current memory: up to half of the population are forcibly mutated artifacts previously created by the agent itself and the rest are generated using ramped half-and-half.

After the population is initialized it goes through the normal iterations of selection and evolution phases. Again, evolution can only use the skills the agent itself possesses. At the end of the evolutionary engine's execution, the agent outputs the best artifact as the product of the agent for that execution. Each agent memorizes all the artifacts itself produces.

4 Collaboration Process

We have devised an experimental collaboration process for two agents creating images with genetic programming. The proposed process is a form of alternating co-creativity [14] with the distinct characteristic that it is executed in turns by independent creative agents. Collaboration allows agents to join their skills to create artifacts, potentially producing outputs with characteristics from both partners. Here we describe the collaboration process and focus on collaboration partner selection in Sect. 5.

The collaboration consists of three consecutive phases:

Initialization phase Together, agents construct an initial population for the collaboration.

Iterative phase Iteratively, agents pass the collaboration population between each other and take turns in evolving it further.

Negotiation phase Together, agents negotiate for the collaboration output.

Next, we describe each collaboration phase in more detail. In the rest of this section, let agents A and A' be the collaborators. If not otherwise stated, the agents use the same parameters and settings when collaborating as when creating solitary artifacts, as described in Sect. 3.

Initialization phase First, the agents must come up with an initial population. To facilitate mutually interesting collaboration, characteristics of artifacts from both agents should be in the initial population. For this, both agents initialize a half-sized subpopulation, using the same initialization method as when creating solitary artifacts. To potentially improve mutual agreement on initial population, both agents pass their newly created subpopulation to each other for the selection. After selection, A combines the subpopulations, and performs crossover and mutation to them to obtain population P. A evaluates P and stores the best artifacts to a hall-of-fame, H (with a maximum capacity of c).

Both agents maintain their own hall-of-fames, H and H', holding the c best artifacts they have observed during the collaboration.

Iterative phase After the initial population has first been evolved into a population P by agent A, P is passed to A'. Then, the collaboration process executes in an iterative manner, much like a typical evolutionary algorithm (A' is the currently active agent):

1. Agent A' evaluates P and stores the best artifacts to its hall-of-fame H'
2. Agent A' applies its selection, crossover and mutation to P, creating P'
3. Agent A' evaluates P' and stores the best artifacts to its hall-of-fame H'
4. Agent A' passes P' to A

After A' has finished its turn, the roles are switched and A executes steps 1-4. The iterative process runs until some termination criterion is met. In our experiments we have used a fixed number of iterations as a termination criteria. After the last iteration, the last active agent passes its last population to another agent for final evaluation.

Negotiation phase The result of our collaboration process is one jointly produced artifact. Therefore, the agents must negotiate on the artifact to choose from the final hall-of-fames, H and H'. If $H \cap H' = \emptyset$, then no collaboration result is produced.

Otherwise, we call collaboration *successful*. Both agents rank the artifacts in their own hall-of-fames from best to worst. The best artifact (b and b') is assigned rank 1, the second rank 2, etc.. After ranking, all artifacts in both of the lists are compared to each other to find matching artifacts. For each matched artifact, we compute *a sum of ranks* using both ranked lists, and choose the artifact with the smallest sum of rank as the result of the collaboration.

5 Modeling Agent's Peers

The problem of peer modeling in collaborative agent societies to facilitate favorable partner selection is the crux of this paper.

Artifact evaluations give an agent direct information of its peers' aesthetic preferences and possibly indirect information about their skills. In principle, learned peer models should capture both properties to maximally benefit an agent in its collaboration partner selection.

In this section we formulate various learning schemes for peer models which we empirically test for their ability to distinguish agents with favorable properties in the rest of the paper. An agent may have different approaches when selecting the collaboration partners, which the learning schemes may model. The approaches can be *altruistic* or *selfish*, or both:

Altruistic approach An altruistic approach gauges how well an agent's artifacts are evaluated by other agents, i.e., it learns which peers appreciate the agent's artifacts most. When used to select collaboration partners, this kind of model selects peers which the agent considers to appreciate most its collaboration partnership.

Selfish approach A selfish approach gauges how well an agent evaluates its peers' artifacts, i.e., it learns which peers the agent itself appreciates most. When used to select collaboration partners, a selfish approach chooses peers which the agent gauges most favorable to itself.

Each agent uses the learned peer models to select its own preferred collaboration partners. It sorts its peers into a preference order using the peer models, and when it is asked to select a collaboration partner, it selects the first peer which is not yet in any collaboration. The peer models enable an agent to be *interaction aware* [6], i.e. the agent monitors and analyses its interaction (artifact evaluations and collaboration results) with other agents and uses that information to control its future interactions by selecting collaboration partners w.r.t. learned models.

Q-learning Q-learning [29] is a common reinforcement learning method, that maintains the expected utility, or Q-value, for each state-action combination based on received reward. It fits our task well, because it can learn continuously from the agent's interactions and it can adapt to the changes in the agent's peers, caused by changes in their memories.

To model an agent's peers, we use a simplified version of Q-learning, that only has one state and for each peer a Q-value, $Q(a)$, where a is the action of selecting a specific peer as collaboration partner. This simplified version also does not take into account possible future rewards. $Q(a)$ is maintained with the update rule $Q(a) \leftarrow Q(a) + \lambda(r - Q(a))$ [22], where r is the received reward and λ the learning rate. We set λ to 0.9 in all of our simulations.

We experiment with three different variations of the Q-learning model called *direct-Q*, *hedonic-Q* and *altruistic-Q*. They differ from each other only in which

artifact evaluations they use for learning. In the rest of this section agent A is the one learning to model its peer A'.

Direct-Q uses as reward the evaluation that agent A gives to the artifact created in collaboration with agent A'. The method learns how much agent A appreciates the artifacts produced in collaboration with agent A', i.e. it is a selfish approach. This is the most straightforward learning scheme in our paper, as it directly models the collaboration results. The disadvantage of direct-Q is that the agent can only learn from doing the actual collaboration, which in many cases restricts the amount of information available for model learning.

To balance exploration and exploitation in direct-Q, we use the ϵ-greedy strategy [21]. The ϵ-greedy strategy selects the best action with probability $1 - \epsilon$ and a random action with probability ϵ. In our context the action is to select the collaboration partner in each collaboration step. We use $\epsilon = 0.2$.

Hedonic-Q learns from artifacts created by its peers alone. The reward is the evaluation A gives to an artifact created by A'. This corresponds to learning which peer A' creates artifacts that are valuable to A making it a selfish approach. Even though hedonic-Q is more indirect than direct-Q, it allows the agent to base its peer model on artifact evaluations. The agent does not have to allocate any resources to artifact creation, which often leads to expansion in information available for learning when more agents are requesting feedback from A than it is able to collaborate with. As hedonic-Q does not gain from ϵ-greedy strategy in our experiments, it does not use it.

Altruistic-Q updates itself based on the evaluations its peers give to artifacts created by A. In other words, altruistic-Q learns how much peers appreciate the artifacts created by A. It has similar advantages over direct-Q as hedonic-Q, i.e. the expansion in information available for the learning. It also does not gain from ϵ-greedy strategy.

Linear regression We use linear regression [30] to investigate if an agent with only one feature (the aesthetic measure) is able to exploit the information from its own initial artifact population in the collaboration partner selection.

Agent A has a linear regression model for each of its peers A'_j. Each model has a weight w_j for the value of A's aesthetic measure and a bias term b_j. A makes a prediction of A'_j's evaluation for an artifact with $\hat{y}_j = w_j x + b_j$, where x is A's observed aesthetic value for the artifact.

The value of selecting A'_j as the collaboration partner is then the sum of predicted evaluations of the artifacts in A's initial population $val(A'_j) = \sum_i \hat{y}_{ji}$, where $\hat{y}_{ji}$ is the prediction for the ith artifact in the initial population.

This can also be formulated as $val(A') = w_j X + c b_j$, where $X = \sum_i x_i$ and c is the amount of artifacts in the initial population, which is constant. From this formulation it is evident, that the value of choosing A'_j is a weighted sum of how much A values its initial population (X) and a constant (c).

We use gradient descent with loss function $\frac{1}{2}(y - \hat{y})^2$ to update the parameters: $w_j \leftarrow w_j - \lambda x(\hat{y} - y)$ and $b_j \leftarrow b_j - \lambda(\hat{y} - y)$, where y is the evaluation given by agent A'_j, $\hat{y}$ is the prediction made by A and λ is the learning rate, which we set to 0.2 in all of our experiments. The parameters are updated when A receives

an evaluation of its own artifact from A' and when A receives an artifact created by A' (evaluation by A' is sent too), making it a hybrid approach as it has both selfish and altruistic influences.

6 Experiments

The main goal in our experiments is to compare the effects of different learning schemes for intentional collaboration partner selection. For each scheme we are interested in the following:

1. What is the overall success ratio for the collaborations? That is, how often did any two agents manage to jointly create a new image.
2. How do agents with different aesthetics differ in their collaboration results?
3. How agents explore the available collaboration partners and do they converge to specific partners?
4. Are agents able to distinguish peers with different skill sets?

We compare the learning schemes to a baseline scheme where agents select their collaboration partners randomly. For each learning scheme, we run the same experimental setup. Each experiment consists of 30 simulation runs and we report the simulation run averages.

The basis of our experiments is an iterative simulation $S = (s_1, s_2, \ldots, s_{200})$. Each simulation involves 20 agents, 4 agents with each aesthetic: Benford's law (BLW), entropy (ENT), global contrast factor (GCF), fractal dimension (FRD) and symmetry (SYM). While agent-specific aesthetics are constant over runs, each agent's skills are sampled randomly anew for each simulation (see Sect. 3).

Within a simulation run, at uneven time steps each agent creates a solitary artifact and at even time steps agents collaborate in pairs. That is, each agent creates exactly 100 solitary and a maximum of 100 collaborated artifacts (if all collaborations are successful). At each time step, after all artifacts have been created, agents send their artifacts to others for feedback, and then update their peer models.

An agent memorizes *all* artifacts it has taken part in creating and may memorize artifacts created by its peers. For peer artifacts, we use memorizing thresholds 0.4 for novelty and 0.5 for normalized value[1].

Agents determine their collaboration partners in each collaboration iteration as follows. First, agents are arranged into a random order. The first agent then pairs up with its favorite partner. We call agents selecting their partners as *selectors*. Then the next free agent acts as a selector and pairs up with its most preferable free agent. This is repeated iteratively until all agents are in a collaboration.

For collaboration process, we use 10 iterations for the iterative phase (i.e., 5 iterations per agent) so that collaborated and solitary artifacts have roughly

---

[1] Normalized value is the value of the artifact (using the agent's aesthetic measure) divided by the value of the artifact the agent has valued the highest during that simulation run (clamped to 1.0).

Table 3. Average collaboration success ratios and various average value measures for learning schemes.

Measurement	Direct-Q	Hedonic-Q	Altruistic-Q	Linear reg.	Random
Collaboration success ratio	0.834	**0.936**	0.864	0.803	0.763
Average value of artifacts					
– Own solitary	0.632	0.634	**0.636**	0.635	0.634
– Selector collaboration	0.580	**0.620**	0.583	0.541	0.551
– Partner collaboration	0.614	0.636	**0.654**	0.653	0.603
Relative value of artifacts per aesthetics					
– BLW own collab./random	1.062	**1.139**	1.120	1.060	1.000
– ENT own collab./random	1.033	**1.102**	1.074	1.044	1.000
– FRD own collab./random	1.058	**1.139**	1.120	1.101	1.000
– GCF own collab./random	1.051	**1.156**	1.147	1.138	1.000
– SYM own collab./random	0.962	0.894	1.014	**1.019**	1.000

the same amount of resources spent on them. Further, for each collaborator's hall-of-fame we use $c = 20$, i.e. it holds 20 best artifacts from the collaboration.

7 Results

In this section we report statistical results from our experiments.

Overall collaboration success and value The average collaboration success ratio and various value measures for each scheme are shown in Table 3. We observe from the table, that each learning scheme increases the overall success ratio of the collaboration (first row). Hedonic-Q has the best success ratio, and altruistic-Q has higher ratio than direct-Q. Linear regression seems to struggle when compared to other learning schemes. On a closer look, altruistic-Q and direct-Q are quite close to each other, which is surprising as direct-Q has 1/19 of the learning information when compared to altruistic-Q (one collaboration per 19 peer evaluations).

The rows three and four in the Table 3 show that learning schemes are able to produce more value (for either selector or partner, or both) in collaboration than the baseline. Further on, there is an interesting bias in the collaboration results. On average, selector values the collaboration results less than the selected partner. Notably, linear regression struggles to produce value to the selector, the selector's average value is even lower than in random, while having high value for the partner. The hedonic-Q gives most value to the selector, but it is still lower than the partner's value. We return to this general bias during discussion. Lastly, in rows 5-9 we observe that all aesthetic measures, excluding symmetry, are on average able to produce more value in their collaborations in any learning scheme than in baseline scheme.

Partner exploration and convergence The collaboration partner exploration is shown in Fig. 2 (left). Hedonic-Q and altruistic-Q are more conservative in their exploration than other schemes. Direct-Q's behavior in relation to random
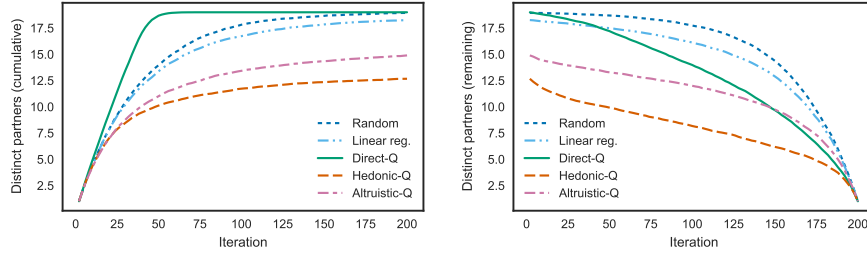
Fig. 2. Average number of distinct collaboration partners accumulated during the simulation run (left) and remaining in the rest of the simulation (right) for each scheme.

scheme is explained by ϵ-greedy strategy's systematic exploration. The convergence is shown on the right in Fig. 2. Especially hedonic-Q has only a few distinct collaboration partners towards the end of the simulation. Notably, direct-Q quickly excludes some of its collaboration partners and its convergence towards the end is more pronounced than altruistic-Q's. (However, due to ϵ-greedy strategy direct-Q does not truly converge in any point.)

The convergence of hedonic-Q is caused by agents collaborating almost exclusively within the same aesthetic (Fig. 3, second from left). Altruistic-Q (Fig. 3, second from right) seems to have converged also to less extent. Direct-Q (Fig. 3, first on left) agents favor peers with same aesthetics, but the effect is not as eminent. Lastly, we see that collaboration between different aesthetics is more common in linear regression (Fig. 3, first on right).

Agent skills We move on to consider if the peer models are able to capture different image creation skills of agents. First, we computed the average number of different skills between each agent pair and average number of different skills in all collaborations and collaborations with their favorite partner. However, there was no meaningful difference between any of them.

Further on, for each individual skill, we computed the average collaboration success ratios and collaboration counts when both collaborators, either selector
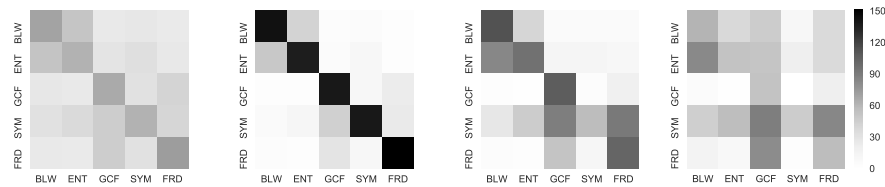

Fig. 3. Heat maps of average collaboration counts between aesthetic pairs for all learning schemes. The aesthetic of an agent selecting the collaboration partner is shown on left and the aesthetic of the selected partner is shown on the bottom of each heat map. From left to right: direct-Q, hedonic-Q, altruistic-Q, and linear regression.
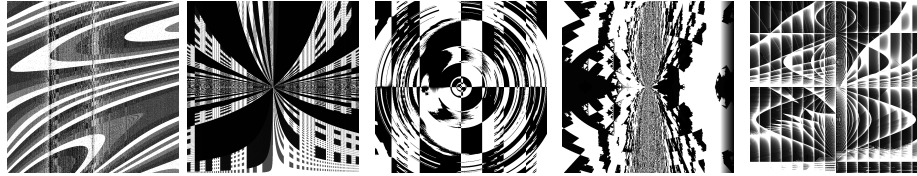
Fig. 4. Selected collaboration results. From left to right: ENT-BLW, FRD-SYM, GCF-GCF, GCF-SYM and SYM-ENT.

or partner, or neither possessed the skill. We could not observe any significant variations between the skills. We return to the possible reasons for these observations in discussion.

Collaboration process For our experimental collaboration process, we observe from our experiments that the process typically creates images where the partners' values for the collaborated artifacts are between their values for artifacts created by either agent alone. For example, for the collaboration pair BLW-FRD we measure that BLW gives values (0.674, 0.510, 0.414) for its own, collaborated (with FRD) and FRD's artifacts. Similarly, FRD gives values (0.580, 0.696, 0.765) to BLW's, collaborated and its own artifacts. This trend is present for every pair of aesthetics. Selected collaboration results are shown in Fig. 4.

8 Discussion and Conclusions

We have described an agent model for creating images with genetic programming where an agent draws inspiration for new artifacts from its memory. We have proposed an iterative collaboration model for such agents and have conducted experiments for intentional collaboration partner selection in societies of agents with diverging skills and aesthetics. From our results, we observed that the learned peer models facilitate successful collaboration when used to select collaboration partners, but also cause convergence to collaborate within similar aesthetics.

Next, we discuss various points of views to and characteristics of our results.

Learning schemes Learning schemes considered in this work fall broadly into selfish and altruistic approaches: direct-Q and hedonistic-Q are selfish, and altruistic-Q and linear regression are altruistic. From the creativity point of view, selfish approaches can be seen as means to enhance agent's own p-creative behavior [10], and altruistic approaches try to empower their collaboration partner's p-creative behavior.

The learned peer models are able to distinguish agents with same aesthetic measures. However, especially for hedonic-Q, the strong preferences over agents with similar aesthetics causes agents to converge to collaborate within the same aesthetic alone. Previously, emerging communication cliques of curious agents sharing similar preferences have been observed by Saunders and Gero [15].

In general, the high collaboration success of the hedonic-Q is not surprising. The final collaboration result is negotiated using the agents' own fitness

functions, favoring selfish approaches. On the other hand, the linear regression's observed behavior is dividing. Having only one feature (the aesthetic measure), the linear regression might not have had enough information to have any real benefit over the simpler Q-learning schemes. Even still it was comparable to altruistic-Q in producing value for the collaboration partner.

It can be argued that for creative agents the most auspicious collaboration partners are not the ones with the exact same skills or preferences, but instead peers with reasonably different capabilities and views [8]. However, the preferences must be similar enough for the agents to be able to agree during the collaboration process [7]. We envision that with more dynamic agent models, e.g. when agent's aesthetic preferences change over time, the convergence would not be as eminent. Moreover, we believe that the online learning methods described in this work would be apt to model such situations.

Aesthetic measures and skills In general, our results between different aesthetic measures in collaboration are in line with the results of den Heijer and Eiben [4]. For example, Benford's law and entropy tend to collaborate quite often in our experiments while den Heijer and Eiben show that they tend to give high values to each others artifacts.

Peer models learned from artifact evaluations alone do not seem to be able to distinguish between agents with different skills. We believe, that this is at least partially caused by the used aesthetic measures. For example, entropy can be satisfied with one skill: a function generating white noise.

To further explore properties of intentional collaboration in creative agents, the aesthetics used should capture a wider variety of artifact properties. For example, by combining multiple simple aesthetic functions together.

Collaboration Process Our experimental collaboration process allows agents to combine their skills and aesthetics to produce artifacts which might not be produced by either agent alone, similar to collaboration in storytelling by Pérez y Pérez et al. [9]. In an ideal situation, collaboration allows an agent to be transformationally creative in its creative process (with another agent) [10]. When analyzing collaborated artifacts, we observe that they have the possibility to preserve characteristics from both agents' creative niches, but in many cases do not. This may be a defect in the experimental collaboration process itself, or be related to the complicated nature of multi-objective evolutionary art [4].

Moreover, the visible bias in collaboration artifact values between selector and partner may be explicable by several features of the process. These include: partner gets to select first from the combined population and to generate the last population, and the elitist selection strategy used in our experiments.

Collaboration is desirably a (super)additive process. Together, the collaborators are able to produce more value or novelty than individually. Unfortunately, the proposed collaboration process is typically a trade-off for the agents: either both agents see the artifact as mediocre, or only one of the collaborators values the result.

For more productive collaboration, the agents must communicate their goals and assumptions, and negotiate future paths for the collaboration. This calls

for more elaborate interaction capabilities and peer models. We believe that the current work serves as an opening for such models and processes.

Overall, peer modeling is a crucial aspect in creative agent societies. Without peer models, there can not be truly social interaction between different agents [12]. In future, we hope to build upon this work to define more sophisticated peer models, and induce social agents with generator awareness [6] to be used in conjunction with learned peer models for more informed collaboration processes.

References

1. Saunders, R., Bown, O.: Computational social creativity. Artificial Life 21(3), 366–378 (2015)
2. Sims, K.: Artificial evolution for computer graphics. In: Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques. pp. 319–328. New York, NY, USA (1991)
3. Romero, J., Machado, P. (eds.): The Art of Artificial Evolution: A Handbook on Evolutionary Art and Music. Natural Computing Series, Springer Berlin Heidelberg (2007)
4. den Heijer, E., Eiben, A.: Investigating aesthetic measures for unsupervised evolutionary art. Swarm and Evolutionary Computation 16, 52–68 (2014)
5. Salehie, M., Tahvildari, L.: Self-adaptive software: Landscape and research challenges. ACM Transactions on Autonomous and Adaptive Systems 4(2), 14:1–14:42 (2009)
6. Linkola, S., Kantosalo, A., Männistö, T., Toivonen, H.: Aspects of self-awareness: An anatomy of metacreative systems. In: Proceedings of the Eight International Conference on Computational Creativity. pp. 189–196. Atlanta, Georgia, USA (2017)
7. Paulus, P., Nijstad, B.: Group Creativity: Innovation through Collaboration. Oxford University Press (2003)
8. Uzzi, B., Spiro, J.: Collaboration and creativity: The small world problem. American Journal of Sociology 111(2), 447–504 (2005)
9. Pérez y Pérez, R., Negrete, S., Nalosa, E.P., Ávila, R., Castellanos, V., Lemaitre, C.: Mexica-impro: A computational model for narrative improvisation. In: Proceedings of the International Conference on Computational Creativity. pp. 90–99. Lisbon, Portugal (2010)
10. Boden, M.: The Creative Mind. Abacus, London (1992)
11. Ventura, D.: Mere generation: Essential barometer or dated concept? In: Proceedings of the Seventh International Conference on Computational Creativity. pp. 17–24. Paris, France (2016)
12. Castelfranchi, C.: Modelling social action for AI agents. Artificial Intelligence 103(1), 157–182 (1998)
13. Csikszentmihalyi, M.: Society, culture, and person: A systems view of creativity. In: Sternberg, R.J. (ed.) The Nature of Creativity: Contemporary Psychological Perspectives, pp. 325–339. Cambridge University Press (1988)

14. Kantosalo, A., Toivonen, H.: Modes for creative human-computer collaboration: Alternating and task-divided co-creativity. In: Proceedings of the Seventh International Conference on Computational Creativity. pp. 77–84. Paris, France (2016)

15. Saunders, R., Gero, J.S.: How to study artificial creativity. In: Proceedings of the Fourth Conference on Creativity & Cognition. pp. 80–87. Loughborough, UK (2002)

16. Eigenfeldt, A., Bown, O., Casey, B.: Collaborative composition with creative systems: Reflections on the first musebot ensemble. In: Proceedings of the Sixth International Conference on Computational Creativity. pp. 134–141. Park City, Utah (2015)

17. Velardo, V., Vallati, M.: On the stylistic evolution of a society of virtual melody composers. In: Proceedings of the Fourth International Conference on Evolutionary and Biologically Inspired Music, Sound, Art and Design. pp. 249–260. Copenhagen, Denmark (2015)

18. Lewis, P.R., Chandra, A., Faniyi, F., Glette, K., Chen, T., Bahsoon, R., Torresen, J., Yao, X.: Architectural aspects of self-aware and self-expressive computing systems: From psychology to engineering. Computer 48(8), 62–70 (2015)

19. Maher, M.L., Merrick, K.E., Saunders, R.: Achieving creative behavior using curious learning agents. In: AAAI Spring Symposium: Creative Intelligent Systems'08. p. 40–46. Technical Report SS-08-03, Stanford, California, USA (2008)

20. Grace, K., Maher, M.L., Mohseni, M., y Pérez, R.P.: Encouraging p-creative behavior with computational creativity. In: Proceedings of the Eight International Conference on Computational Creativity. pp. 120–127. Atlanta, Georgia, USA (2017)

21. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction, vol. 1. MIT press Cambridge (1998)

22. Claus, C., Boutilier, C.: The dynamics of reinforcement learning in cooperative multiagent systems. In: Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence. pp. 746–752. Madison, Wisconsin, USA (1998)

23. Albrecht, S.V., Stone, P.: Reasoning about hypothetical agent behaviours and their parameters. In: Proceedings of the 16th Conference on Autonomous Agents and Multiagent Systems. pp. 547–555. São Paulo, Brazil (2017)

24. Machado, P., Cardoso, A.: Computing aesthetics. In: Proceedings of the Advances in Artificial Intelligence: 14th Brazilian Symposium on Artificial Intelligence (SBIA'98). pp. 219–228. Porto Alegre (1998)

25. Vinhas, A., Assunção, F., Correia, J.a., Ekárt, A., Machado, P.: Fitness and novelty in evolutionary art. In: Proceedings of the Fifth International Conference on Evolutionary and Biologically Inspired Music, Sound, Art and Design. pp. 225–240. New York, NY, USA (2016)

26. Romero, J., Machado, P., Santos, A.: On the socialization of evolutionary art. In: Proceedings of the Applications of Evolutionary Computing: EvoWorkshops 2009. pp. 557–566. Tübingen, Germany (2009)

27. Zhou, A., Qu, B.Y., Li, H., Zhao, S.Z., Suganthan, P.N., Zhang, Q.: Multiobjective evolutionary algorithms: A survey of the state of the art. Swarm and Evolutionary Computation 1(1), 32–49 (2011)

28. Tanese, R.: Distributed genetic algorithms. In: Proceedings of the Third International Conference on Genetic Algorithms. pp. 434–439. San Francisco, CA, USA (1989)

29. Watkins, C.J., Dayan, P.: Q-learning. Machine learning 8(3-4), 279–292 (1992)

30. Murphy, K.P.: Machine learning: a probabilistic perspective. MIT press (2012)