Negotiating Gestalt: Artistic Expression by Coalition Formation Between Agents

Kaye Mason, Jörg Denzinger, and Sheelagh Carpendale

Department of Computer Science, University of Calgary, Canada
{katherim, denzinge, sheelagh}@cpsc.ucalgary.ca

Abstract. We present a system using semi-autonomous agents to help artists express ideas. Agents control their representation on a canvas via interactions in agent space. They are given rules, which include the ability to form, join, leave and dissolve coalitions. While individually, agents constitute only small parts of the composition, together they form more and more complex parts until the full picture emerges. The artist can take direct control of an agent or coalition if the result is unsatisfactory.

We have implemented the *Surreal* engine that realises these ideas and we present a case study of its usage in producing Mondriaan-like paintings.

1 Introduction

In expressing ideas, emotions and opinions, artists often use media that consist of fundamental elements composited into a *gestalt* – a whole that is greater than the sum of the parts. For composers the elemental pieces can be notes, for writers – words, and for painters – strokes on canvas. One way of describing this is to see the style of individual artists as self-defined rules of how elements are grouped together, and an individual composition as created through the development, application and refinement of these rules. In this paper, we consider the implications of this in the medium of computer graphics. A computer graphics image has been taken as a collection of graphical elements, combined through rules provided by an artist working with an algorithm. We take this a step further, and consider these graphical elements as self-interested agents in a multi-agent system. The *gestalt* in our paradigm comes from the *emergent behaviour* of these agents as they seek to follow a set of artist-specified rules.

A practical challenge rests in the sheer numbers of image-elements that can be required to compose an image. One solution has been to write algorithms that deterministically place image-elements on a base image or model, leaving the artist to focus on the creation of this image or model while the algorithm does the image-element placement. The drawback of this approach is the algorithmic dominance of the style of the final composition. While it is acceptable that tools influence a work in any medium, it is important to continue to search for a better balance between artistic expression and algorithmic support.

By applying multi-agent techniques – taking each image-element as a semi-autonomous agent, and the composition as an expression of a multi-agent system – we find a new way of seeing an old problem. Still, it is not immediately

A. Butz et al. (Eds.): SG 2005, LNCS 3638, pp. 103–114, 2005.

apparent how it is easier for the artist. Not only do artists have to work with large numbers of static elements, but now they have large numbers of agents, each with its own views about what its environment should look like. It is only when we introduce coalition formation that the power of this approach becomes apparent.

When image elements form coalitions, they band together into larger groups that an artist can manipulate, making the tasks of building an image from elements more reasonable. Instead of dealing with masses of tiny elements, the artist can deal with a smaller number of coalitions (groups) of elements. The process of creating an image becomes the process of training agents about the conditions under which coalitions should form, and the properties they should have. Conceptually, the *gestalt* becomes the grand coalition, and the composition is complete when it is achieved to the satisfaction of the artist.

2 MAS-Coalitions for Artistic Expression – A Framework

Computers have provided artists not only with a new medium but also a range of possibilities and a detail of control that can be overwhelming. We first look at state-of-the-art computer graphics tools and philosophies behind them. Then we present the framework for our multi-agent based approach.

2.1 Artistic Expression Tool Philosophies

Many of the computational tools developed for artists involve an algorithm, which adds elements to a composition to alter its appearance. Strothotte et al. presented a general framework for this process in [12], and since then many different artistic styles have been turned into algorithms, including watercolour painting [3], expressionist, pointillist and impressionist painting [4,5], and pencil sketching [1,2]. All of these techniques have something in common – they involve adding algorithmically placed elements to an artist-defined image or model.

Initiated by Smith [11], researchers are also considering how to best deal with such graphical elements. Smith's models are defined by a formal grammar, and he uses the term *graftals* to describe his grammar based graphical elements. Smith's ideas were extended by Kowalski et al. [7]. In these models, Smith's notion of a grammar-based element is extended to include procedurally-based image elements, which are placed over the final image at points determined to be appropriate by the algorithm. These ideas have been extended by Markosian [9] and by Kaplan [6].

2.2 Agents and Coalitions for Artistic Expression

Our multi-agent framework for aiding artistic expression follows in the tradition of graftals, but extends it to give a picture element a more active role, even to making it nearly autonomous. Picture elements pursue goal fulfillment and the final composition or *gestalt* results from emergent agent behaviour. In the following, our framework is described in terms of agents, the agent space, and the emergent gestalt resulting from coalition formation.

The Agents: Within our framework, an agent $\mathcal{A}g = (h, R, S, T)$ consists of a happiness function h, a set of rules R governing its behaviour, a set S that contains the agent's possible subjective spatial understandings and a set T with the possible values of the agent's internal data tables. The function h is selected by the artist to express what makes the agent happy. It is the goal of each agent to optimize its individual happiness. The set of rules R consists of behavioural rules R_{Beh} and rules R_{Exp} that guide how the agent expresses itself on the canvas ($R = R_{Beh} \cup R_{Exp}$). The agent selects the behavioural rule that will optimise its happiness by evaluating the possible resulting elements from S. If no rule increases happiness, then no behavioural action is taken. The rules in R_{Exp} deal with how the agent manifests on the canvas. All the rules in R_{Exp} that can be applied will be applied in addition to the selected rule from R_{Beh}.

The Agent Space: Agents exist in a space that is full of other agents each striving for their own idea of what constitutes happiness. However, in this graphics application the term *space* is significant. The space in which we view images we will call *canvas space*. Our conceptual framework of agents that make decisions about their own expression requires an additional spatial abstraction. These agents do not exist in the canvas space, but in *agent space*. From agent space, it must be possible for the agents to create a representation in canvas space. The set S of possible spatial understandings of an agent represents the agent's subjective view of both spaces.

Since artists often use and reuse picture elements in different contexts and the same picture element may occur thousands of times in a given image, it should be possible to define an agent for a particular picture element once. To support this, we introduce the concept of an agent *tribe*. Agents of a tribe start with the same h-function, the same sets R, S, and T, but they can start with different actual values for S and T.

Coalitions and the Emergent Gestalt: Coalitions form when an agent discovers that the fulfillment of its goals is compatible with goals of other agents. Our coalitions, in common with other MAS (see, for example [13]), operate like individual agents, $\mathcal{C}o = (h, R, S, T)$. Rich variation in coalition formation is possible and necessary in our application to support expression. $\mathcal{C}o$ can be arrived at through: construction from the corresponding components of the member agents, influence from a new agent member, or use of a predefined set. Due to the artist's ability to plan and interact, we can allow coalitions to become members in other coalitions, and agents and coalitions can even be members of several coalitions.

In addition to the types of rules agents have, coalitions also have rules that deal with accepting additional agents and with dissolving a coalition. Individual agents can leave coalitions; however, if an agent selects to leave, it can make sense to dissolve a coalition. For example, when a coalition contains only one other agent or when the artist finds a coalition unsatisfactory. This requires that such rules are treated specially and they are executed without considering happiness.

3 The *Surreal* Engine

The artist is an important aspect of our system. If our agents behave badly, interference is possible (and should take place). This solves some issues, but also presents challenges. While the principles of our framework apply equally to writing or to music, the *Surreal* engine is presently aimed at visual arts. Therefore we must acknowledge and respect the skills of visual artists. These rely largely on visual feedback, and we must provide that feedback in reasonable time, and without overwhelming detail. We call this the Visual Principle.

3.1 Defining Agent Space

Our agent space is topologically equivalent to a hyper-torus — a cube with periodic boundary conditions. This decision was made for simplicity, to allow us to easily use existing techniques in MAS research which operate on that basis. If an agent leaves one face of the hyper-torus, it reappears at an opposite position. This simplifies agent movement — the agents cannot move outside valid bounds — they just move and deal with their new environment.

Given the large number of agents involved, it can be hard for one agent to find others. If each agent has to compare its position with all others to find neighbours, computation time is $\mathcal{O}(n^2)$. This inhibits prompt feedback. To address this, we introduce a coordination artifact after Omicini et al. in [10] – a dynamic spatial grid in agent space. We use a grid, as opposed to other methods of dividing space (octrees, BSP-trees, etc), because we have no guarantees of continuity in the density of agent space, and the grid does not require any maintenance as this shifts over simulation steps. That said, it is possible that in future work, this function could be offered in a better way.

When an agent is placed or moves, it registers with the grid. Each grid cell keeps a list of registered agents in its territory. An agent can query its own cell or neighbouring cells for a list of registered agents. We allow the artist to set the grid resolution dynamically if desired, to address agent density issues, and to prevent the algorithmic degeneracy that occurs if all agents are in the same grid cell.

3.2 Defining Agents

As already discussed, we use tribes to define similar agents with a prototypical agent. Here, we consider prototypical agent definitions in four parts, based on the tuple which defines an agent in our framework. For each, we look at specifics of our implementation. For ease of consideration of practical issues, we rearrange the discussion slightly from the last section.

Agent Spatial Understanding (S): All agents in the *Surreal* engine begin with no neighbourhood understanding. They know that they exist and they understand the nature of agent space. They know that there may or may not be other agents, and they know how to perform searches for these agents, and how to move themselves in agent space. Using this, an agent can find others and share information, including tribal membership, data tables, and even spatial understanding. In general, agents have the ability to see any other agent. It is possible, however, to exclude an agent from noticing agents with particular properties. Likewise, it is possible for an agent to choose not to register with its grid cell, making it invisible to other agents performing grid searches.

Data Tables (T): Data tables can hold any information the artist wants the agent to have, within the limits of system memory. The tables can be referenced by the agent's rules in R (see below). This data may or may not be freely visible, depending on the simulation. Some examples of data which may be stored are:

- The agent's position in agent space, as $(p = (x, y, z, w))$.
- Visual representations for the agent as images, models, or drawing methods.
- Transformation $M = M_{Translate}(t_x, t_y, t_z) \cdot M_{Rotate}(\rho, \phi, \theta) \cdot M_{Scale}(s_x, s_y, s_z)$.
- Critical values for an agent's happiness $(\alpha_0 - -\alpha_n)$ in the range $0 \leq \alpha_i \leq 100$.
- Special data that is held by the agent for the particular simulation.

Generally, data tables are built up as a simulation runs – forcing the artist to deal with it can be overwhelming, a violation of the Visual Principle. However, it is possible for the artist to interfere, and set or adjust values if they wish to get that deeply involved.

Agent Happiness (h): We represent the agent's happiness as a percentage. No happiness is 0%, while a state of perfect happiness is 100%. Depending on the simulation, an agent may begin with no happiness, with perfect happiness, or with something in between. Every simulation step, each agent adjusts its happiness by checking for relevant factors. In many simulations, it is useful to decrement each agent's happiness by a small amount in every simulation step. This forces the agents to continue to actively improve their situation.

Rules (R): The *Surreal* engine supports the expression and behaviour rules outlined in the last section as follows:

Expression Rules use information from an agent's data tables (T) to represent that agent in *canvas space.* No conflict resolution is applied — in every simulation step, each display rule that can be executed is executed. This may result in the agent having no representation or multiple representations.

- *Universal Rules* are always applied. They include a reference to a visual representation, a transformation matrix, and possibly other items from the agent's data tables (see above).
- *Conditional Rules* are rules of the form $A \rightarrow B$, where A is a condition that can be tested, and B is an expression rule, formed like a universal rule. The predicate A could be a conjunction, formed as $A = a_0 \wedge a_1 \wedge ... \wedge a_n$ $a_0, ..., a_n$ are some predicates defined on either agent or canvas space and tested on the agent's subjective understanding.

Behaviour Rules can rely on information in the agent's spatial understanding (S), as well as in the agent's data tables, as follows:

- *Exploration Rules* involve the agent gathering data and exploring agent space. These rules often modify the agent's spatial understanding (S) either by gathering new information or by choosing to forget information.
- *Movement Rules* change the position of the agent in agent space. We represent this in homogeneous coordinates (a position tuple $p = [x, y, z, w]$), stored in the agent's data tables. Movement rules alter these values. The agent can also report its positional change to the agent space grid, so its registration can be updated. Agents that are deliberately hiding from other agents would not do the latter.
- *Relationship Rules* alter an agent's understanding of its relationships with other agents. These can include rules involving the formation and breakup of coalitions, in requesting information from another agent, or in answering requests for information.

3.3 Defining Coalitions

When defining and forming coalitions, we use an encounter-based approach, like Lerman [8]. Because there is no agent with an overall view, agent decision making is performed in a bottom-up manner. That said, the agent's understanding of its own goals is heavily influenced by the artist. The artist may provide rules which instruct the agent that it would be well-served to form a coalition with agents with particular properties. Still, an agent's decisions are always based on what information it has collected. We have generally opted to use one agent coalition's director. While it is possible to use voting schemes or other methods within the *Surreal* engine, these can consume resources and prevent us providing the prompt feedback required by the Visual Principle.

Because we allow an agent to belong to as many coalitions as it finds advantageous, conflicts arise. In practice, this does not happen often, because the simulation specifications are planned with care. However, when conflicts do arise, an agent must leave one of the conflicting coalitions. Since coalitions, once formed, can represent themselves as an agent in the system, we define them to have the same components as an agent. More formally, we define a coalition as a tuple $\mathcal{C}o = (h, R, S, T)$, as follows:

Coalition Spatial Understanding (S): A coalition begins with the spatial understanding of all its members. This allows the coalition to make decisions based on information that is at least as good as its members would have on their own. If a coalition $\mathcal{C}o = \{\mathcal{A}g_0, \mathcal{A}g_1, ..., \mathcal{A}g_n\}$ then $S_{\mathcal{C}o} = \{S_{\mathcal{A}g_0} \cup S_{\mathcal{A}g_1} \cup ... \cup S_{\mathcal{A}g_n}\}$.

Since we use directed coalitions, all agents share their understanding $S_{\mathcal{A}g}$ with the director. This information is left in the repositories of the individual agents. Copying the data to a different location takes time, which prevents the system from offering adequate visual feedback. When an agent leaves a coalition, it takes its spatial understanding $S_{\mathcal{A}g}$ with it, and when a new agent joins, its understanding is automatically added to $S_{\mathcal{C}o}$.

Coalition Data Tables (T): The director of the coalition has access to the data tables of all members. Again, if a coalition is $\mathcal{C}o = \{\mathcal{A}g_0, \mathcal{A}g_1, ..., \mathcal{A}g_n\}$, then $T_{\mathcal{C}o} = \{T_{\mathcal{A}g_0} \cup T_{\mathcal{A}g_1} \cup ... \cup T_{\mathcal{A}g_n}\}$. Special data items may be created for the coalition — for example, the coalition's current happiness (h_t).

Coalition Happiness: Coalition happiness is similar to agent happiness. When the coalition comes into existence, its happiness measure is created by the coalition's director, as distinct from that agent's own happiness. No bottom-up approach to coalition formation can ensure optimal or even desirable coalitions. We must allow agents the possibility of leaving or dissolving a coalition. Though agents must follow coalition rules, and can gain and lose happiness through the coalition's actions, they still keep track of their own happiness. If it falls below one of their critical values, an agent will leave the coalition.

A special case of leaving occurs when the happiness of the director, or the happiness of the coalition falls below a critical value. In either case, the director may decide that the coalition is failing, and that it should dissolve. This is similar to Weiß [13]. If this happens, any special data stored by the director for the coalition, like coalition happiness and critical values, is deleted.

Coalition Rules (R): Evaluation of the rules is done by the assigned director that also initiates the execution of the rules.

Expression Rules for coalitions are acceptable in the *Surreal Engine*. They are accomplished in much the same way as for individual agents, and are orchestrated by the director of the coalition. Usually having them means that individual agents only use conditional rules for individual expression, with the condition being an absence of coalition membership.

Behaviour Rules can be divided into movement, exploration, and relationship rules, as with agents. The form of these rules will depend on the simulation, however, in our simulations, we have found some generalities are useful. While in a coalition, members generally agree to give up any actions involving independent movement, unless instructed by the coalition's director. Exploration by individual agents can continue as before, and honest agents share new information with a coalition's director, although cheating agents may not. Agents in a coalition can continue to form relationships with agents outside, and even join multiple coalitions. If there is a conflict in agreements, they will leave whichever coalition has been giving them the least happiness.

4 Session – Piet Mondriaan

To illustrate the workings of our system we have chosen a simulation that imitates the style of the artist Piet Mondriaan. The inherently mathematical aspects of Mondriaan's abstract compositions allow us to create a straight forward set of rules for demonstrative purposes.

It should be noted that instead of trying to create our own artistic style, we are actually attempting to imitate the style of an existing artist, as a proof of concept. We have a number of original simulations with this system, based on our own ideas and those of other artists, but the simplicity here allows us to focus on the system, and not the details of the simulation:

4.1 Setup and Definitions

In order to set up the fundamentals for Mondriaan, the artist has specified four tribes of agents, each of which contains a piece of information required to draw a primitive to canvas space, and three specialised coalitions. All of these have some common elements:

- *Happiness Function.* The happiness of all agents and coalitions is defined in the range $0 \leq x \leq 100$, the lower bound being no happiness, and the upper perfect happiness. The happiness of the agent in the current time step (h_t) is calculated by $h_t = min(max(h_{t-1} - 1, 0) + h_{act}, 100)$, where h_{t-1} represents the happiness at the last time step, and h_{act} represents happiness gained by particular actions. Some values for happiness increases could be: joining a coalition: 50; getting a new member: 25; showing off on canvas: 1.
- *Despair Condition.* We define critical values $\alpha, \beta, \gamma, ...$ for each agent and coalition (such that $\alpha \leq \beta \leq \gamma$), beyond which an agent will take particular actions. The lowest of these (α) is a special value, below which an agent will enter a state of despair. In this state, it will accept any coalition offer it gets. In this way, we encourage agents to join the gestalt grand coalition, even if they have not been able to enter the intermediate coalitions.
- *Distribution Procedure.* At the start, agents of all four tribes are distributed randomly across the canvas in numbers that have been previously specified by the artist.

Orientation Tribe: Agents in this tribe provide later line coalitions with an orientation, i.e., horizontal or vertical (in T). The two orientations are approximately evenly distributed across agents. In addition to holding this information, orientation agents act as directors of line coalitions. They have no expression rules, and six behaviour rules. These rules move the agent in agent space until a coalition possibility is found, form line coalitions and announce this formation. They record other line coalitions (in S) to track the next drawing location. They may also reach a despair condition, as described above.

Width Tribe: These agents provide line coalitions with line width information (in T). The width values are distributed across agents such that there are few lines at the widest width, and more at the narrowest. They have no expression rules and four behavioural rules which direct them to move in agent space until they are invited into a line coalition, or reach their despair condition.

Offset Tribe: Offset agents provide line coalitions with information about how far to offset from the last line drawn of the same orientation (in T). This allows the creation of wide, unused portions of canvas, as favored by Mondriaan. These agents have no expression rules, and four behaviour rules which provide the same functionality as the width tribe's rules.

Colour Tribe: These agents provide fill coalitions with a colour (in T). They also direct fill coalitions, and are responsible for recruiting members. They have no expression rules, and five behaviour rules, which involve movement in agent space, recruitment of line coalitions, and the despair condition.

Line Coalitions: A complete line coalition consists of an orientation (director), a width and an offset agent. They have one universal expression rule – to express as a line with a width specified by the width agent, an orientation specified by the orientation agent, and offset from the last line drawn of the same orientation, or the edge, in the absence of any line. They have four behaviour rules, moving in agent space until invited to join a fill coalition by a colour agent, and a despair condition. A line coalition may join multiple coalitions, as long as no two of these coalitions result in adjacent fill areas.

Fill Coalitions: A complete fill coalition consists a colour agent (director), and two line coalitions of each orientation. The coalition's expression is then a rectangular fill of the area between these four lines. This expression may overlap other line coalitions. If so, it is drawn on top of them. The fill coalition has five rules, related to movement and exploration in agent space, the formation of gestalt coalitions, and the despair condition.

Gestalt Coalitions: A gestalt coalition is the final attempt of agents and coalitions to stabilise their environment. It is directed by the first agent or coalition to propose its formation. Complete fill coalitions will always accept a proposal to join this coalition. Other agents which have reached their despair condition will always accept a proposal to join. Another gestalt coalition will always accept a proposal to join, and give control to the director that made the request. Gestalt coalitions use the happiness function h_t modified by the fact that their existence allocates one happiness percentage to each member in each time step. They have no expression rules, and no behaviour rules – the rules for recruiting new members are contained within the rules for a fill coalition.

4.2 Execution

We will not show all steps in the execution, only check points at which something crucial occurs. We also do not show artist interaction – human computer interaction is a complex issue for this application, as with any application, and the primary concern of this paper is with the MAS aspects.

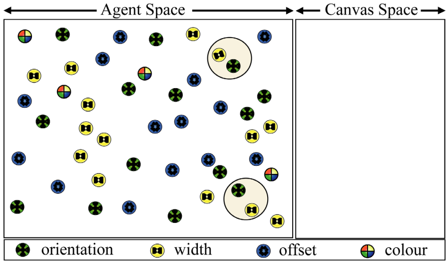

Fig. 1. First coalitions appear

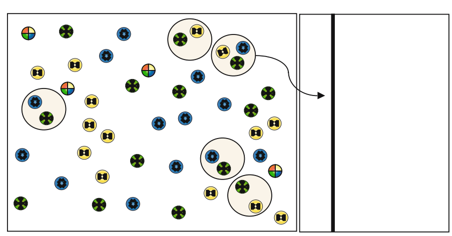

Fig. 2. First line coalition

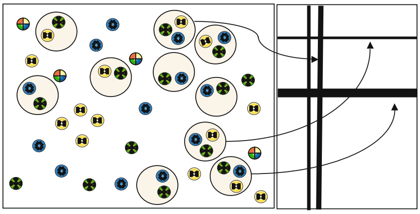

Fig. 3. More line coalitions

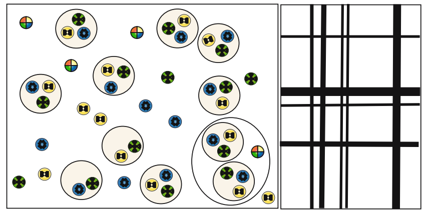

Fig. 4. Budding fill coalition

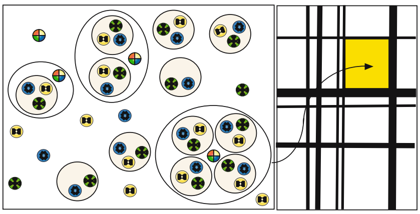

Fig. 5. First fill coalition

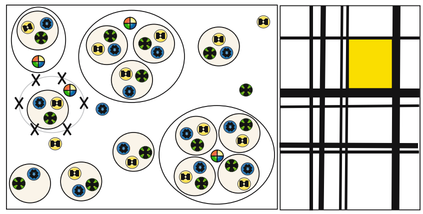

Fig. 6. Fill coalition dissolves

At the start of the simulation, agents are scattered over agent space, unaware of their surroundings. They begin to come together in coalitions, but none yet has the requisites to express itself in canvas space. See Figure 1. Figure 1 also presents the legend for all the figures.

The first coalition of three agents from the right tribes appears, and a corresponding line is drawn in canvas space with the width, offset and orientation specified by the agents in the new coalition. See Figure 2.

More coalitions meet the requisites, and more lines are drawn in canvas space with the offsets, widths and orientations specified by their members. See Figure 3.

Coalitions of line coalitions begin to form. At this point, however, no line contains the requisite pair of coalitions of each orientation, so no fills can be drawn. See Figure 4.

The first fill coalition appears when four line coalitions find one another. There are two horizontal and two vertical line elements, and on the canvas the fill is drawn between the lines represented by these elements. See Figure 5. Fill coalitions continue to form. At this point, the colour agent directing one fill coalition decides the coalition is not beneficial, and dissolves it. See Figure 6.

New fill coalitions appear, including fill coalitions sharing members. Two coalitions which share one line coalition in agent space will share one line in common in canvas space. See Figure 7. The coalitions of fills begin to ally with one another and with free agents and line coalitions. Together, they begin to form the gestalt coalition. Once in this coalition, the happiness of each agent will grow to perfect. See Figure 8.

The grand coalition is achieved. At this point, the composition is complete, unless the artist wishes to interfere with it, to break coalitions, or to make changes to individual agents. See Figure 9.

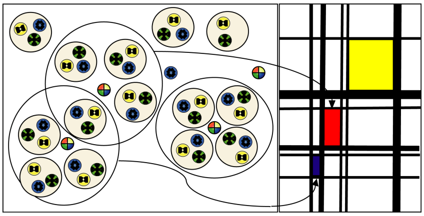

Fig. 7. More fill coalitions

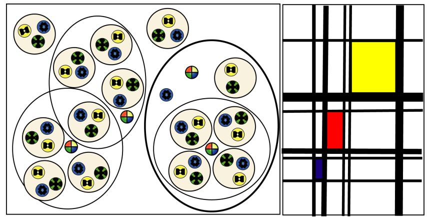

Fig. 8. Gestalt coalition begins

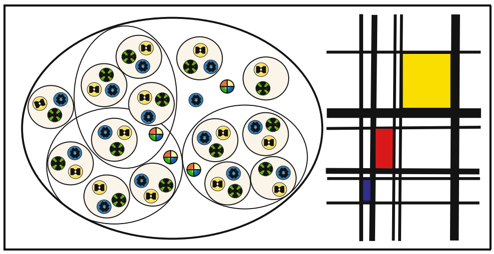

Fig. 9. Gestalt grand coalition

Note that every time we run *Surreal* with these agents, we get a different composition, because the initial placement and data distribution across agents is randomised. As a result, the coalitions develop differently.

5 Conclusions and Future Work

We have presented a framework for the application of multi-agent systems and coalition formation to the process of creating artwork. We have shown how the *Surreal* engine uses this framework, using a simple case example – defining agents and coalition types for art similar in style to the works of Piet Mondriaan.

The basic concepts we propose allow for various instantiations and while *Surreal* already offers some choices, more of them will have to be added to cater to more styles. Figure 10 shows an image in the Japanese *seigaiha* style created with Surreal. The use of these techniques in the areas of music, writing, and

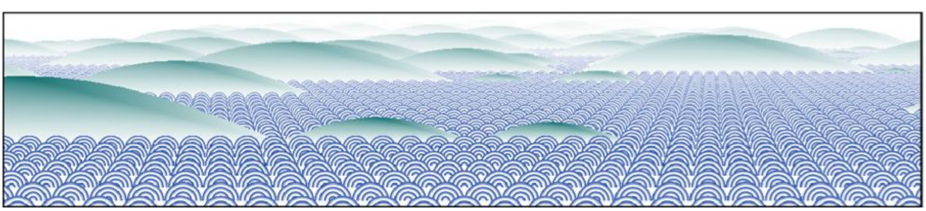

Fig. 10. Seigaiha pattern done with agents using this framework

other arts remains to be developed, but we believe that the same techniques we have used could be applied in those problem domains, and we look forward to tackling these areas in the future.

Acknowledgments

Research supported by the Natural Sciences and Engineering Research Council of Canada and the Alberta Informatics Circle of Research Excellence.

References

1. M. Costa Sousa and J. W. Buchanan. Computer-generated graphite pencil rendering of 3d polygonal models. In *Eurographics*, 1999, pp 199–207.
2. M. Costa Sousa and J. W. Buchanan. Observational model of blenders and erasers in computer-generated pencil rendering. In *GI*, 1999, pp 157–166.
3. C. J. Curtis, S. E. Anderson, J. E. Seims, K. W. Fleisher, and D. H. Salesin. Computer-generated watercolor. In *SIGGRAPH*, 1997, pp 421–430.
4. A. Hertzman. Painterly rendering with curved brush strokes of multiple sizes. In *SIGGRAPH*, 1998, pp 453–460.
5. A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. . Salesin. Image analogies. In *SIGGRAPH*, 2001, pp 327–340.
6. M. Kaplan, B. Gooch, and E. Cohen. Interactive artistic rendering. In *NPAR*, 2000, pp 67–74.
7. M. A. Kowalski, L. Markosian, J. Northrup, L. Bourdev, R. Barzel, L. S. Holden, and J. F. Hughes. Art-based rendering of fur, grass, and trees. In *SIGGRAPH*, 1999, pp 433–438.
8. K. Lermann and O. Shehory. Coalition formation for large-scale electronic markets. In *ICMAS*, 2000, pp 167–174.
9. L. Markosian, B. J. Meier, M. A. Kowalski, L. S. Holden, J. Northrup, and J. F. Hughes. Art-based rendering with continuous levels of detail. In *NPAR*, 2000, pp 59–66.
10. A. Omicini, A. Ricci, M. Viroli, C. Castelfranchi and L. Tummolini Coordination Artifacts: Environment-Based Coordination for Intelligent Agents In *AAMAS*, 2004, pp 286-293.
11. A. R. Smith. Plants, fractals, and formal languages. In *SIGGRAPH*, 1984, pp 1–10.
12. T. Strothotte, B. Preim, A. Raab, J. S. Mann, and D. R. Forsey. How to render frames and influence people. In *Computer Graphics Forum* 13, 1994, pp 455–466.
13. G. Weiß. *Distributed Machine Learning*. Infix-Verlag, Sankt Augustin, 1995.