

AISB '03
Cognition
in Machines
& Animals

**Proceedings of the
AISB '03 Symposium on
Artificial Intelligence and Creativity
in Arts and Science**

7th – 11th April 2003
The University of Wales, Aberystwyth

EPSRC



Prifysgol Cymru
Aberystwyth
The University of Wales



The AISB'03 Convention
COGNITION in MACHINES and ANIMALS

7th - 11th April 2003
University of Wales, Aberystwyth

www.aish.org.uk

Proceedings of the Symposium on
Artificial Intelligence and Creativity in
Arts and Science

Published by

The Society for the Study of Artificial Intelligence and the Simulation of Behaviour

www.aisb.org.uk

ISBN 1 902956 34 X

Printed at the University of Wales, Aberystwyth, SY23 3DB, Wales

The proceedings of the five symposia in the AISB'03 Convention are available from SSAISB. They are entitled as follows:

The Second International Symposium on Imitation in Animals and Artifacts
ISBN 1 902956 30 7

The Third Symposium on Adaptive Agents and Multi-Agent Systems
ISBN 1 902956 31 5

Scientific Methods for the Analysis of Agent-Environment Interaction
ISBN 1 902956 32 3

Biologically-Inspired Machine Vision, Theory and Application
ISBN 1 902956 33 1

Artificial Intelligence and Creativity in Arts and Science
ISBN 1 902956 34 X

The AISB'03 Convention

Cognition in Machines and Animals

The Society for the Study of Artificial Intelligence and the Simulation of Behaviour (SSAISB) is the UK's national society in the field of Artificial Intelligence. It is one of the oldest AI societies in Europe and has a long track record of supporting the UK research community. The society has a tradition of holding annual meetings and a convention format has proved successful, most recently at the Universities of Edinburgh (1999), Birmingham (2000), York (2001), and Imperial College, London (2002). We are delighted to host the 2003 convention at the University of Wales, Aberystwyth.

Recent advances in artificial intelligence, cognitive science, neuroscience and robotics have stimulated new interest in the relationships between biologically-based and computer-based cognitive systems. There is a current upsurge of activity both in biologically-targeted computer modelling and in adapting and experimenting with biological concepts for the design and development of novel computing technologies. For example, the BBSRC and EPSRC recently started a joint programme on Adaptive and Interactive Behaviour of Animals and Computational Systems (AIBACS). These topics are providing much inspiration for new research and innovation and the general theme of the 2003 convention reflects this stimulus and has attracted a range of high quality paper submissions and presentations.

In the early years of the SSAISB there was considerable interdisciplinary excitement about the general phenomenon of cognition, and lively meetings were attended by both computer scientists and psychologists. Perhaps the success of AI technology across a wide range of hard applications in the '80s and '90s caused the research focus to drift away somewhat from that most fascinating and astounding source of inspiration: cognition in the human brain. Consequently, it is very pleasing that we are now in the throes of a swing back towards those original concerns and I hope this convention will encourage further interaction between psychology and computer science, not least through the exciting selection of topics and invited speakers.

The convention is structured around five independently organized symposia, each with their own Programme Chairs and Programme Committee. The proceedings of all symposia are published and several of the best papers have been selected for edited publication in the AISB Journal. The unifying theme of the convention is reflected in a series of plenary sessions which feature invited keynote speakers. The six speakers, all international authorities in their field, are: Professor Peter Hobson, from the Tavistock Clinic and University College London; Professor Rajesh Rao, from the Center for Mind, Brain and Learning, University of Washington; Professor Giulio Sandini, from LIRA-Lab, University of Genova; Professor Sorin Solomon, from Theoretical Physics, Hebrew University of Jerusalem; Professor Yiannis Aloimonos, from the Computer Vision Laboratory, Institute for Advanced Computer Studies, University of Maryland; and Professor Gregor Schöner, from the Institut für Neuroinformatik, Ruhr-Universität Bochum.

We are extremely grateful for the support of the EPSRC, who provided funds for both the invited speakers' travel costs and for a large number of bursaries for postgraduate research students. The bursaries cover most of the cost of attending the convention and are particularly valuable as they enable the next generation of scientists to interact and to discuss issues with some of the key players in their research field. Support and assistance has been given by the Department of Computer Science at UWA, Meinir Thomas and the Conference Office at UWA, the IEE, and the Committee of the SSAISB, especially Geraint Wiggins. Special thanks are

due to all the symposia Chairs for all their time and hard work: Kerstin Dautenhahn, Chrystopher Nehaniv, Dimitar Kazakov, Daniel Kudenko, Eduardo Alonso, Ulrich Nehmzow, Gregor Schöner, Horst Holstein, Fred Labrosse, Pablo Gervas and Simon Colton. Special thanks also to our Technical Programme Assistant, Joanne Walker, and, for typesetting expertise, to Fred Labrosse. Finally, thanks to all the authors for their contributions, which form the essential core of the event. I am looking forward to the programme and I wish everyone a stimulating and productive meeting.

Mark Lee (Convention Host and Programme Chair)

Presented here are the proceedings from the fifth AISB symposium on creativity in AI and Cognitive Science. This builds on similar events in 1999 (Edinburgh) 2000 (Birmingham) 2001 (York) and 2002 (Imperial). The purpose of these symposia has been to bring together researchers in all aspects of creativity research within AI and cognitive science. Along with philosophical discussions about the nature of creativity in both arts and sciences, we have particularly tried to encourage the presentation of research discussing implemented systems and frameworks within which implementations can be undertaken, assessed and applied.

The aim of holding one unified meeting, instead of several simultaneous smaller ones, is to promote communication between those studying different aspects of creativity. We are very pleased to say that computational creativity attains more respectability and attention every year and is beginning to flourish into a very interesting research area.

In this year's proceedings, the research presented covers philosophical approaches to machine creativity, including a study of epistemic and ethical mediators and the functioning of tropic communication. Frameworks for machine creativity are also covered with papers on optimality principles for conceptual blending, and the use of concept maps for natural language generation. As usual, computer music is well represented at the symposium, with papers discussing macrostructure discovery for musical signals, driving chord progressions using tension, fugal exposition composition, improving the prediction of PPM variants, music generation from statistical models and an approach to improvisation using multi-swarms. Finally, computer art motivated by human perception is represented with a paper on symbolic pointillism.

The papers presented here were carefully selected by multiple blind anonymous peer review. This, as always, entailed a lot of work for the symposium's programme committee, to whom we are very grateful. They were:

Amilcar Cardoso, University of Coimbra, Portugal

Graeme Ritchie, University of Edinburgh, UK

Geraint A. Wiggins, City University, UK

John Gero, University of Sydney, Australia

Ashok Goel, Georgia Tech, USA

Alison Pease, University of Edinburgh, UK

Marcus Pearce, City University, UK

We are also very grateful to the following people who reviewed papers for this symposium: Andres Melo, Carola Boehm, Francisco Camara Pereira, John Barnden, Kia Ng, Penousal Machado and Peter Desain. Finally, we wish to thank the authors for presenting high quality research and for their punctuality and cooperation in preparing this volume. We are also very grateful to the convention organisers for making the event possible.

Pablo Gervas

Universidad Complutense de Madrid, Spain

Simon Colton

Imperial College, London, UK

Contents

The Functioning of Tropic Communication: A Mechanism for Consistent Figurative Descriptions for Artistic Effect. <i>D. Billinge and T. Addis</i>	1
Optimality Principles for Conceptual Blending: A First Computational Approach. <i>F. C. Pereira and A. Cardoso</i>	9
Creative Abduction as Active Shaping of Knowledge: Epistemic and Ethical Mediators. <i>L. Magnani</i>	20
Music Generation from Statistical Models. <i>D. Conklin</i>	30
Symbolic Pointillism: Computer Art motivated by Human Perception. <i>N. Krüger and F. Wörgötter</i>	36
Swarm Music: Improvised Music with Multi-Swarms. <i>T. M. Blackwell</i>	41
Natural Language Generation from Concept Blends. <i>F. C. Pereira and P. Gervás</i>	50
A connectionist approach to driving chord progressions using tension. <i>A. F. Melo and G. Wiggins</i>	60
A n-gram approach to fugal exposition composition. <i>A. Craft and I. Cross</i>	68
An empirical comparison of the performance of PPM variants on a prediction task with with monophonic music. <i>M. Pearce and G. Wiggins</i>	74
Computational creativity by structural analogy between acoustic bayesian models. <i>M. A. Casey</i>	84

The Functioning of Tropic Communication: A Mechanism for Consistent Figurative Descriptions of Artistic Effect

Dave Billinge

Department of Creative Technology,
University of Portsmouth,

Buckingham Building, Burnaby Road, Portsmouth PO1 3AE
email: dave.billinge@port.ac.uk

Tom Addis,

School of Computer Science and Mathematics,
University of Portsmouth,

Buckingham Building, Burnaby Road, Portsmouth PO1 3AE
Visiting Research Fellow Science Studies Centre,
Department of Psychology, University of Bath
email: tom.addis@port.ac.uk

Abstract

We describe our investigation into the consistency of response to music by two or more people. We explain why we view these responses as private and knowable only by the respondent. Our current position is that people communicate using tropic language. This model of the discourse is shown diagrammatically starting with a sample of the type of language studied. The functional model constructed from these diagrams in a schematic programming language is described. With this model we show that communication is possible in principle with private referencing. One conclusion is that such private referencing could be an underlying mechanism for a creative act.

1. Introduction

The basic question we are exploring here is, 'Is it possible for people to have a mutually consistent response to the world given only internal and private references?' This paper describes the conceptual and computer-modelling work so far and states how we see it developing.

The above question has arisen from our work in trying to analyse the nature of descriptions of musical effect in order to build an artistic decision support system. We discovered from the observation of peoples' language use while they discuss music under controlled conditions that there seemed to be no consistent way in which words were used to describe music. It was only when we considered metaphor that we could make sense of what we had observed. In particular, we noted that speakers used metaphor creatively and it was because of this that the inconsistency was observed. The question then arose as to why people found it necessary to use metaphor. We concluded that it was because the descriptions were about personal feelings. Since feelings are private, the accepted referential mechanism of language cannot take effect.

Since, in addition, we do, routinely, have just such conversations, not only about art, and believe we are communicating successfully, we see the act of communicating musical experience as a microcosm of a much larger issue. The issue is how humans manage to converse when so often they appear not to be saying what they mean.

It is essential to point out what we are not researching. We are not trying to show what music means¹, (if indeed that is a real problem at all), and we do not believe that people need to agree on artistic descriptions as they would need to agree on descriptions of, say, poisonous mushrooms in the forest. However, in trying to show how people can communicate figuratively we do need to establish that some common understanding of the language exists.

2. Tropic Communication

It is the view of some linguists, such as Lakoff and Johnson (1980) that our natural language does not function as a computer language with defined terms and unambiguous reference (formal semantics). Our language is overwhelmingly figurative; it is tropic² communication. If this is the case, and we wish to build machines that work within the same paradigm as people so that emotion can be communicated, then such tropic communication must be understood. However, the process behind such communication seems to depend upon the construction of descriptions from purely internal reference points; namely primary emotions or emotional-archetypal situations. How such a system can work is puzzling, but one way of understanding and exploring the process is to build a computer model.

3. The Conceptual Model

A model that has many of the properties we are looking for and that seems to support the creative construction of metaphorical descriptions of musical effect is conceptual blending. Conceptual blending (Fauconnier and Turner 2002) suggests a way in which different situations (such as given by mixed or simultaneous metaphors) can be brought together to form new perceptions of the world. As with metaphor such perceptions can then be used to infer consequences and thus possible action.

In order to ensure that conceptual blending fulfils the role of describing tropic communication we need to review our observational results (Billinge 2001, Bilinge and Addis 2001, 2002).

¹ Extensive consideration has been given to this issue by Meyers (1956, 1967, 1973), Cooke (1959) and many others. For more considerations see Bilinge (2000).

² A trope is a figure of speech such as metaphor, metonymy and synecdoche.

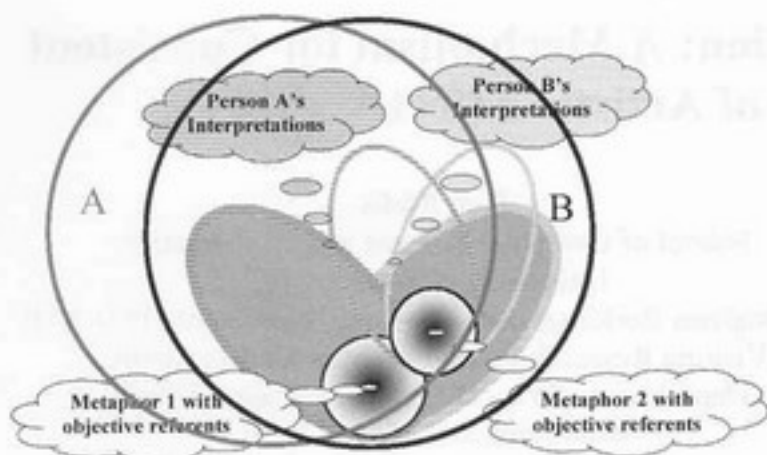


Figure 1: Venn diagram of the emotional landscapes for Actors A and B and the effect of two metaphors

The Venn diagram summarises these results. It is also an indication of the process as we saw it at the time but in a way that leaves too much to the imagination. The diagram is essentially static and does not show all the necessary dynamics of creating descriptions through hidden and private references. What we needed to show was that the internal referencing was a possible basis for communication and it could be stable. We had to determine the strategies needed for it to work and to show the underlying assumptions required.

4. The Functional Model

4.1 Background

In an attempt to address these issues and to detail this process, we are now working on a functional computer model of the internal referencing process. This should lead to a comparatively simple version of a conceptual blending model. It is a dynamic model, which demonstrates the mutually adaptive internal processes that allow tropic communication to work. We hope to use this model to revisit our observations and create new controlled experiments to find evidence for the model and in the light of this to refine it or create a new one.

As a starting point in this new investigation, a text by a music critic was chosen from a music magazine. This text is typical and it is intended to communicate the emotional effect of a piece of music.³

(X's) refreshingly iconoclastic insert notes insist that this performance of the Brahms' Second reveals an 'intrinsic unease' stemming from Horowitz's 'temperamental mismatch' with both Toscanini and the work itself. That may be true but whatever contentiousness clouded the recording session, the heat of the occasion forged it into an interpretative virtue. The result is one of the most arresting recordings of this warhorse ever made. Granted, it's controversial. This is not the place to seek out the succulence of Brahms's harmonies (passages that are poignant in other hands sound nervously expectant here) or the sweetness of his melodies (calls for dolce and espressivo are rebuffed even in the third movement). Rather, Horowitz and Toscanini deploy every available tool to scratch out any signs of tenderness from this normally bighearted score: barbed attacks, gaunt tone, angular phrasing, contestatory treatment of

accompaniments, urgent tempos, highly strung rhythms (dotted figures have a special edge). The climaxes erupt with an unaccustomed violence; and while the finale is notable for its leggiro, it's the lethal lightness of a well-aimed stiletto. It is, in other words, far from easy listening. But despite a few moments of scramble, the performance has such ferocious commitment that it's hard to resist. If the stately account of the Tchaikovsky sounds less heterodox, that's largely because, in my view, the work is more overtly dramatic to begin with. Still, even here, the music is pushed to extremes. Every single bar crackles with tension and Horowitz and Toscanini forgo the music's lyricism in order to wring every possible thrill from the score. It is, without doubt, a cruel sacrifice; but once again, the reading is so electrifying that you're compelled to go along with it.⁴

On the surface this is a set of undefined, maybe poetic, words especially if seen, as we did in our initial experiments, as a discrete lexicon. However if we inspect the text for metaphors a different view emerges. The metaphors may be considered the medium for transmission of the musical experience.

The author sets the stage with a statement of the stance he takes and intends to convey.

"X's refreshingly iconoclastic insert notes insist that this performance of the Brahms' Second reveals an 'intrinsic unease' stemming from Horowitz's 'temperamental mismatch' with both Toscanini and the work itself."

The way he reinforces this message is to write in terms of conflict, violence and tension whilst using harmoniousness as a foil. It would be inappropriate to go into this in overmuch detail so a few examples must suffice. The text is full of conflict metaphors:

Iconoclastic insert notes ... contentiousness clouded the recording session one of the most arresting recordings of this ... it's controversial ... calls for *dolce* and *espressivo* are rebuffed ... deploy every available tool to scratch out any signs of tenderness ... barbed attacks ... gaunt tone ... contestatory treatment of accompaniments ... lethal lightness of a well-aimed stiletto ... you're compelled to go along ...

And at the other end of the emotional scale it has a significant number of metaphors of harmony and pleasure:

the succulence of Brahms's harmonies (passages that are poignant in other hands) ... the sweetness of his melodies ... signs of tenderness from this normally bighearted score ... the music's lyricism ...

These examples are typical of the written genre and indeed of the spoken discourse. Modelling this must involve simplification thus we chose to work in terms of messages being conveyed between exclusive private worlds where the only common factor is the source of the experience and its comparators.

³ A more detailed account of this work was presented by the authors at the ECAI'02 conference in Lyons, France in August 2002. See Billinge and Addis (2002).

⁴ From International Record Review, Vol 3 Issue 5 August 2002: ISSN 1468-5027 p.27

4.2 The Initial Model

We wanted to show how the reactions of an Actor B to a piece of music, as perceived by Actor A, might be quantified to enable processing into an internal model of B held by A. Such a process will involve a form of negotiation and learning that requires the use of tropic communication by both Actors. It should be noted that the weightings we apply are not seen as ‘real’ mental artefacts that exist within the human brain (although they might). Any measures we use are simply convenient representations of emotional intensity within the computational model.

We start by describing how the processes are modelled.

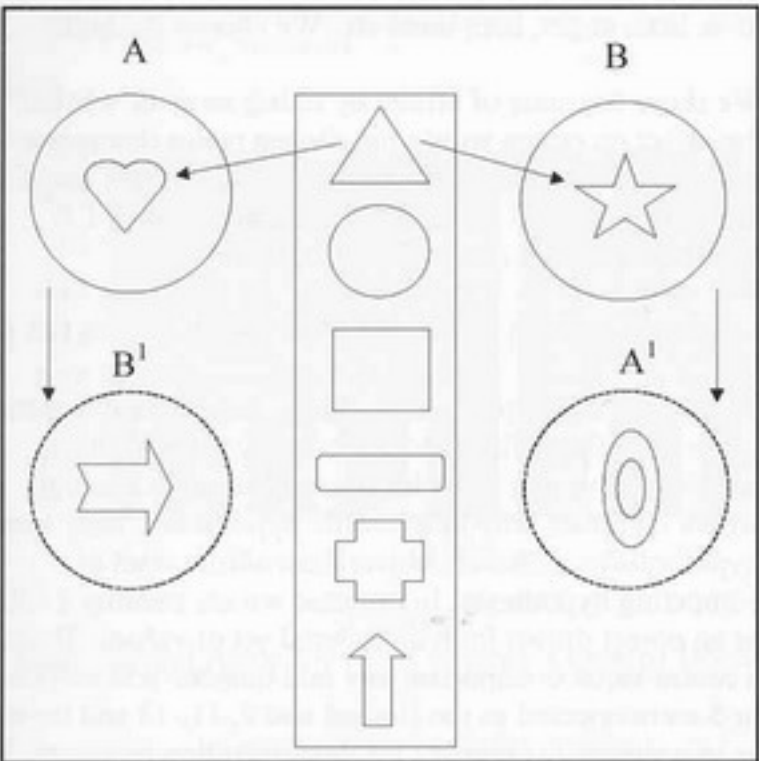


Figure 2: Tropic communication using internal views

Figure 2 illustrates the situation. This diagram shows an external world in the central rectangle that consists of six artistic objects, in this case musical performances (events) of pieces (objects) called Triangle Music, Circle Music etc.

In this world Actors A and B listen to Triangle Music and then A modifies his⁵ internal interpretation of Actor B's description. It is postulated that Actor A's and Actor B's internal emotional reactions differ. This is shown by the different shapes within the hard circle associated with each actor. This leads inevitably to the kind of differences of opinion commonly experienced by those discussing art works. We assume B provides a figurative description in which his (B's) differing perception of the effect of Triangle Music is conveyed to Actor A.

This process is now traced in more detail. As noted, the series of shapes in the central oblong represent different pieces of music experienced by both A and B. They can therefore describe their respective reactions to the first of these, Triangle Music, by comparing the effect of Triangle Music with the effect of Circle Music, Square Music, and so on. This use of comparative reference is what we take to be characteristic of figurative discourse. A's initial emotional reaction to Triangle Music is represented by a heart shape, B's by a star. This implies that both Actors have their own individual responses. In conversation B

describes to A how he feels about Triangle Music. As a result of this explanation, A begins to build an internal representation of what he believes to be B's emotional reaction: this is shown in Figure 2 by the soft circle B¹. A can never know exactly what reaction has taken place in B's mind because we cannot describe our emotional states directly, only indirectly via the use of what we describe as ‘Tropic Communication’, the use of metaphor, simile and other tropes; other figurative language. The inevitable result of this tropic description is that A's picture of what is in B's mind is inaccurate, thus B¹ is shown containing a polygon only loosely similar to the star B is ‘actually’ describing. The only means B can use to describe his reaction to Triangle Music is by making statements like ‘It was more dramatic than Circle Music’, or, ‘It was much less lively than Cross Music’, and so on. We note that ‘dramatic’ and ‘lively’ are emotional dimensions that are understood by each actor.

In parallel with these inputs, A has to interpret B's comments via his own emotional reaction, shown in Figure 2 as a ‘heart’ reaction. Further, when B draws a comparison with Circle Music as above, A's interpretation of the comparison not only has to modify from the basis of a ‘heart’ effect, it also has to be modified using A's and not B's, reaction to Circle Music. So when B says, ‘Triangle Music was more dramatic than Circle Music’, A's internal model B¹ is modified by a statement which could be stated like this. ‘B thinks that the dramatic impact of Triangle Music is more than the dramatic impact of Circle Music so I must alter B¹ to reflect how I perceive the dramatic nature of Circle Music to make it more dramatic than my perception of Triangle Music.’ A has no means whatever of directly observing what B thinks about anything. He has to make a *creative* leap from state B¹ to, perhaps, state B², B³, and so on, until B has finished drawing comparisons with all the available music know to them both. What this model is intended to convey is not only the inaccuracy that must inevitably follow from A trying to picture something he can never see, an image inside B's mind, but also the convergence of the internal models that results from the accumulation of comparisons.

We aim to show how the feedback triggers adjustments of A's emotional reaction weightings and leads to his reassessing his categorisation of the basic emotional states conveyed by his experience of the artistic event. With these states changed, Actor A is obliged to rework his internal descriptions and adjust internal references so that they are acceptable both to himself and to Actor B. This also suggests that without some common external reference (the event) no change of their internal models can take place.

4.3 The Abstracted Model

The extended nature of the above description was reason enough for us to view the further modelling of this interaction by, so far as we are able, refining it down to the bare essentials. Figure 3, below, is still a diagrammatic representation but it provided for us a means of resolving the various objects into numeric measures and program labels.

⁵ The English term ‘his’, as is the traditional syntactic convention, will be considered gender neutral.

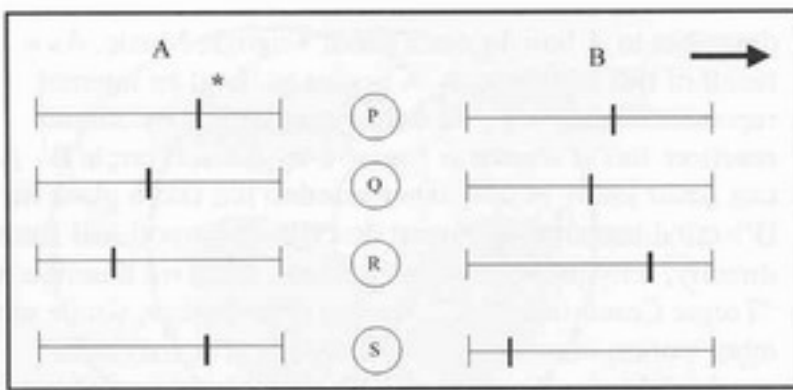


Figure 3: Further Abstracted Model of Tropic Communication

In Figure 3 the music pieces are represented as P to S, the Actors as A and B and the heavy arrow shows the potential for an indefinite number of additional Actors C, D, E and so on. The lines represent one of many simplified single-emotional reactions to Music P, Q, R and S. We call these lines 'dimensions' and they are associated with each external object. For example the extent to which the music P makes Actor A happy is shown by the position of the bisecting vertical line at *, the horizontal line representing the continuum between totally-without-happiness on the left and fully-happy to the right. Thus Music P makes A strongly happy, Music Q makes A moderately happy and so on through R and S. Actor B's happiness quotient is similarly shown and we can see it is different for Music R but quite similar for Music P. Thus when B tells A that Music P makes him feel happier than does Music S, Actor A will interpret that declaration against his internal model of the happiness quotients he has attributed to Music P and Music S. Since A finds P to be less-happy music than S he might well respond to B by expressing surprised disagreement. B, who cannot be aware of A's internal ratings, will come back with a further comparison with which he hopes to clarify the extent of Music P's happiness in his estimation. He might respond, "OK, but surely you find R happier than P." The reader can see that this will not bring agreement since A's internal model still will not be able to support that belief, but it will enhance A's understanding of B's vision of the world.

The use of continua to represent strength of emotion, and statements of sequence to represent the tropic communications which are all that may pass between A and B, allowed us to construct a functional model which would simulate the course of internal model change.

Adding the potential for an Actor C, D, E and so on, allows us to extend the model to simulate another factor, the process by which A may choose with whom he wants most to talk. If A converses with B and C, he will eventually find that one or the other tends towards better agreement with his, A's, assessments. In future exchanges A may choose to take the views of C more seriously because they agreed better last time. To account for this we adopt the notion of belief (Addis and Gooding 1999) as a weighting of C's views in A's estimation. Tropic communication such as this takes place in a social situation and it is our personal experience that such changes in social bonding do, in fact, take place.

4.3 The Basic Functional Model

The current model shows how actors negotiate an agreed internal emotional framework that neither of them can observe directly. This serves in turn as a model of how they also agree choices of metaphor and other tropes as

well as create a figurative description that both accept as an adequate representation of their feelings. We further show how the current model reflects these differing effects as numerical weightings and discuss the types of experiment that we are carrying out to refine the model.

The starting point for Actors 1, 2 and 3 is they have no musical experience having attended no concerts. So we show them as having no knowledge of what we describe above (Figure 2) as Triangle Music, Circle Music, Square Music, and so on. They have no means of communication about the effect of these virtual compositions because they have no reference points. They cannot describe reactions in terms of "more" or "less" (for example) happiness because they have no happiness-concept to begin with. The dimension of experience chosen could be anything, love, hate, anger, happiness etc. We choose the latter.

We show this state of affairs by rating an actor's belief in the effect on others within the chosen realm (happiness can continue to serve) of a piece of music. Since belief must never exceed unity, the maximum value of 1 is distributed evenly over a 7-part scale thus:

$$[0.143 \mid 0.143 \mid 0.143 \mid 0.143 \mid 0.143 \mid 0.143 \mid 0.143]$$

The choice of a 7-part scale may need explanation because we refer to these dimensions as continuous. We have adopted the notion of belief from Addis and Gooding where it applies only to scientific hypotheses. They treat a hypothesis as a discrete object drawn from a set of competing hypotheses. In our case we are treating a value as an object drawn from an ordered set of values. Because a centre value is important any odd number will suffice. 3 or 5 were rejected as too limited and 9, 11, 13 and beyond as unnecessarily complex for demonstration purposes. 7 can be seen as a useable compromise. We use 0 to represent no (happiness) and 6 to represent absolute (happiness). Only the actor themselves knows this measure and for now we will not pursue questions such as "how does he know?"

After experience of a single object our actors can only make absolute statements like "Triangle Music makes me feel (happiness) or (nothing)", so communication between actors 1, 2 and 3 is limited to (happiness present), and (happiness absent). Internally the object will have triggered a degree of (happiness) but it cannot be expressed. In our model we will represent this personal experience as a belief of 1 for one of the seven positions on the continuum because "he knows what he likes".

If we communicate with actor 4 who has never attended any concert, then Actor 1, for example, could tell Actor 4 that the music was of no consequence, or report that he should have the experience. This level of effect is akin to Pierce's "firstness". Actors 1, 2 and 3 have experienced something they can graduate internally but cannot express, but Actor 4 lacks the experience that even allows identification of the dimension.

The actors are now exposed to the second object, Circle Music. A similar process of reception takes place. Owing to the fact that Actors 1, 2 and 3 have one item of prior experience they can now compare that emotional effect with the effect of object two on an identified dimension. This gives them a more subtle means of communication in that they can convey internal comparisons.

To represent the experience of attending a concert (event 1) we instruct our model through queries. Note that all interactions are called “queries” in the schematic functional programming language *Clarity*. (Addis and Townsend-Addis 1995, 1998, 2001, 2002).⁶

```

QUERY> actor_only_concerts (no_query) #1
True
  
```

Each actor, knowing their own response to the objects, then modifies their initial neutral assessment of others views. The function show_views displays all actors’ perception of others for all objects. For clarity we have graphed the results. We have also dropped Actor 1 for simplicity.

```

QUERY> show_views #1
True
  
```

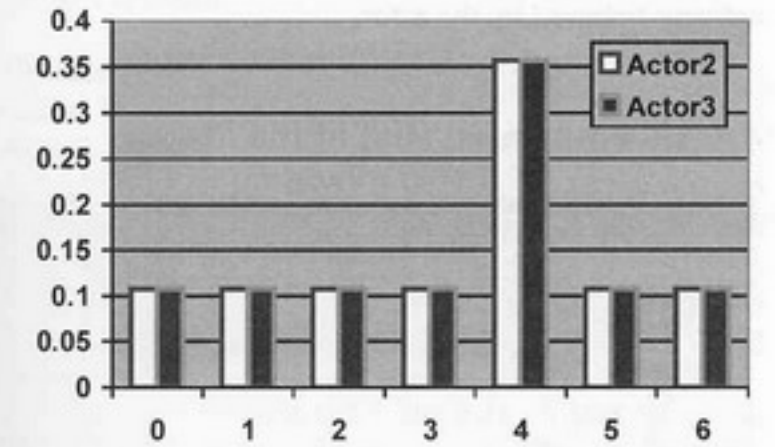


Figure 3
Actor 2’s and Actor 3’s views of other’s view of Object 1

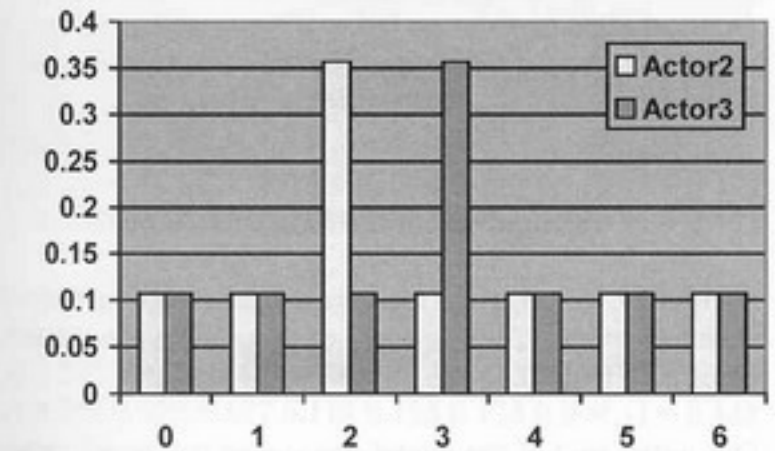


Figure 4
Actor 2’s and 3’s views of other’s view of Object 2

It is initially assumed by all Actors, without any other knowledge, that all other Actors will have the same view as themselves. The distribution of effects has now changed from the base distribution of 0.143 all actors have to an increased rating at one point of the scale by 0.25 to 0.357. Since all of the seven positions on the scale still has to total to unity, the remaining items dropped back to 0.107.

We now illustrate a single communicative exchange between actors 2 and 3 using the function consult_person shown in figure 5.

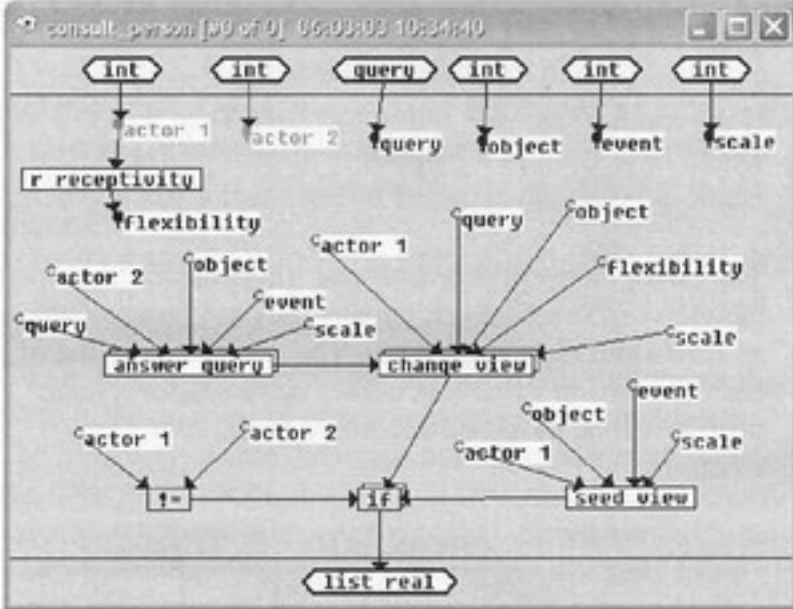


Figure 5
Clarity function for consulting and changing view

This shows two processes: answer_query and change_views. In this example Actor 2 asks Actor 3 “is Object 2 of higher value than Object 1?” In this case the response is “No”. This then triggers change_view for Actor 2, which is graphed in Figure 6. The change can be seen in relationship to Actor 3’s view that remains unchanged.

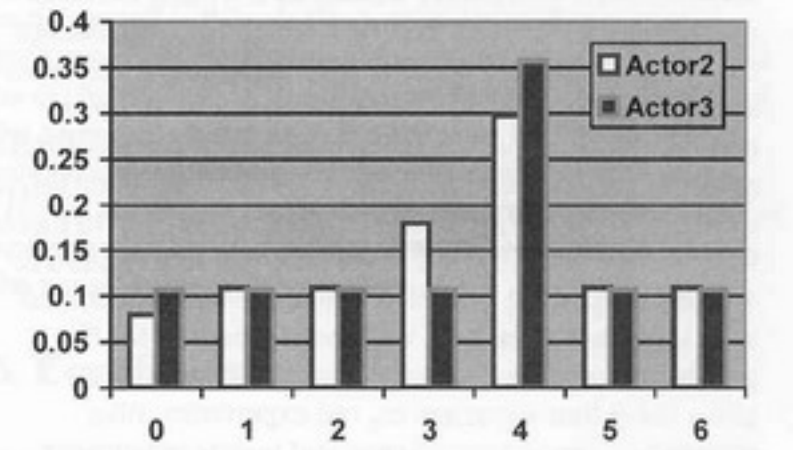


Figure 6
Object 1 modified by Object 2 for Actor 2.

To be more specific, when Actor 2 asks Actor 3 if Circle music (object 2) is liked better than Triangle music (object 1), he gets the answer “No”. In response, Actor 2 modifies his internal distribution of Triangle music (object 1) by shifting his own distribution of Circle music (object 2). He shifts it to be higher than the current position, using it to modify his vision of Actor 3’s view of Triangle music (object 1). Note that in figure 6 the belief in Object 1 (Triangle music) is now higher, in position 3, than shown in figure 3. This is because Object 2 (Circle music) was high for Actor 2 in position 2, which was then shifted right by one position. The extreme left value for the modification was taken as zero.

In the functional language Actor_2 asks Actor_3 if object 2 is more than object 1 as follows:

```

QUERY> answer_query (more #2) #3 #1 #1 #1
False
  
```

The distribution of 2’s view of object 2 is shifted right. This distribution is used to modify the distribution of 2’s view of 3’s view of Triangle music (object 1).

⁶ A free version of ClarityLite is available at www.clarity-support.com

Experience

Actor 1

1[0.107 0.107 0.107 **0.357** 0.107 0.107 0.107] I = 0.165

2[**0.357** 0.107 0.107 0.107 0.107 0.107 0.107] I = 0.165

Actor 2

1[0.107 0.107 0.107 0.107 **0.357** 0.107 0.107] I = 0.165

2[0.107 0.107 **0.357** 0.107 0.107 0.107 0.107] I = 0.165

Actor 3

1[0.107 0.107 0.107 0.107 **0.357** 0.107 0.107] I = 0.165

2[0.107 0.107 0.107 **0.357** 0.107 0.107 0.107] I = 0.165

Without this experience, no changes or modifications are possible. The values that have a belief greater than the indifference threshold are highlighted.

Eight cycles of consultation take place where each Actor asks all other actors a sequence of questions concerning their own experiences. The replies are used to modify each Actor's distribution of beliefs on some view of the group's opinions:

Consult

(more 1)

Actor 1

1[0.107 0.107 0.107 **0.357** 0.107 0.107 0.107] I = 0.165

2[**0.253** 0.113 **0.235** 0.113 0.113 0.113 0.060] I = 0.158

Actor 2

1[0.107 0.107 0.107 0.107 **0.357** 0.107 0.107] I = 0.165

2[0.113 0.113 **0.253** **0.235** 0.113 0.113 0.060] I = 0.158

Actor 3

1[0.107 0.107 0.107 0.107 **0.357** 0.107 0.107] I = 0.165

2[0.113 0.113 0.113 **0.376** 0.113 0.113 0.060] I = 0.171

(more 2)

Actor 1

1[0.060 **0.178** 0.113 **0.310** 0.113 0.113 0.113] I = 0.161

2[**0.253** 0.113 **0.235** 0.113 0.113 0.113 0.060] I = 0.158

Actor 2

1[0.060 0.113 0.113 **0.178** **0.310** 0.113 0.113] I = 0.161

2[0.113 0.113 **0.253** **0.235** 0.113 0.113 0.060] I = 0.158

Actor 3

1[0.060 0.113 0.113 0.113 **0.376** 0.113 0.113] I = 0.171

2[0.113 0.113 0.113 **0.376** 0.113 0.113 0.060] I = 0.171

(same 1)

Actor 1

1[0.060 **0.178** 0.113 **0.310** 0.113 0.113 0.113] I = 0.161

2[**0.211** 0.123 **0.197** 0.114 0.128 0.128 0.099] I = 0.148

Actor 2

1[0.060 0.113 0.113 **0.178** **0.310** 0.113 0.113] I = 0.161

2[0.132 0.128 **0.207** **0.192** 0.114 0.128 0.099] I = 0.148

Actor 3

1[0.060 0.113 0.113 0.113 **0.376** 0.113 0.113] I = 0.171

2[0.132 0.128 0.128 **0.276** 0.109 0.128 0.099] I = 0.152

(same 2)

Actor 1

1[0.091 **0.164** 0.122 **0.239** 0.127 0.127 0.129] I = 0.149

2[**0.211** 0.123 **0.197** 0.114 0.128 0.128 0.099] I = 0.148

Actor 2

1[0.097 0.127 0.121 0.159 **0.239** 0.127 0.129] I = 0.149

2[0.132 0.128 **0.207** **0.192** 0.114 0.128 0.099] I = 0.148

Actor 3

1[0.097 0.127 0.127 0.116 **0.276** 0.127 0.129] I = 0.152

2[0.132 0.128 0.128 **0.276** 0.109 0.128 0.099] I = 0.152

In the final modification after seven more cycles we have:

Actor 1

1[0.089 **0.167** 0.131 **0.198** 0.136 0.138 0.142] I = 0.146

2[**0.180** 0.132 **0.191** 0.126 0.138 0.139 0.094] I = 0.146

Actor 2

1[0.093 0.139 0.133 0.160 **0.198** 0.136 0.142] I = 0.146

2[0.145 0.137 **0.172** **0.186** 0.126 0.139 0.094] I = 0.146

Actor 3

1[0.093 0.138 0.138 0.127 **0.227** 0.135 0.142] I = 0.148

2[0.145 0.138 0.135 **0.227** 0.121 0.139 0.094] I = 0.148

Given the limited experience of each Actor this shows that there is the possibility of convergent internal standards.

The limitation is that the upper bounds of the evaluation of the objects cannot be shifted until new and 'better' objects have been experienced. Actor 1 has inferred correctly the relative position of object 2 with respect to object 1 and Actor 2 has adopted an evaluation of object 2 that is approaching a consensus.

5. Future Development

This initial experiment with this model was only intended to establish the possibility of inferred internal referencing. Experiments that are more extensive will need to be done to show what might be the limitations and stability of such internal and inferred references. More Actors experiencing a wider range of objects over an extended set of events is required. Investigations into the way in which the internal referencing might change are needed as well as the seeing if there is the possibility of distinct sub-groups of Actors appearing due to drift. Such drift would be expected as soon as the Actors become independent of the forced consultation sequence used here. The model will be extended along the lines of Addis and Gooding (1999). We intend to use the development of this model to revisit our observations. We would expect this to lead to the construction of new experiments.

6. Conclusions

We have shown within the limitations of this model that internal referencing can have some external meaning. This relies firmly on the common humanity of the individual actors, in that they have to share a common cognitive structure. This study also suggests that without some common external reference with which this cognitive structure may interact no communication can take place.

Moreover the results suggest that more experience and interaction would produce internal models that are more congruent. Despite this potential for convergence there could remain important differences between individuals. It is these differences that could be the source of creative acts since perceived novelty occurs because the observer lacks the internal logical structures to construct the observed act. This implies that creativity is a result of individual differences remaining despite social interaction. Because we do share a common cognitive structure it should not be surprising that some people will develop common internal views at the same time. This would result in new insights occurring independently and simultaneously within a particular culture. This should not suggest that people need isolation to be creative, quite the reverse; people need to be in communication in order for novelty to be recognised and accepted.

References

- ADDIS T. R. AND TOWNSEND ADDIS J. J. (2002) 'An Introduction to Clarity: A Schematic Functional Language for Managing the Design of Complex Systems.', Vol 56, No4, April, International Journal of Human Computer Studies, ISSN 1071 5819, pp331-422.
- ADDIS T. R. AND TOWNSEND ADDIS J. J. (2001) 'Avoiding Knotty Structures in Design: Schematic Functional Programming.', International Journal of Visual Languages & Computers, 12, Pages 689-715, ISSN 1045 926X.
- ADDIS T. R. AND GOODING, D. C. (1999) 'Learning as Collective Belief-Revision: Simulating Reasoning about Disparate Phenomena.' AISB Symposium, 6-9 April, University of Edinburgh. Pages 19-28, ISBN 1 902956 04 4
- ADDIS T. R. AND TOWNSEND ADDIS J. J. (1998) 'A Functional Schematic Interpreter: an environment for Model Design '. International Journal of Systems Research and Information Science, Vol. Pub Gordon Breach Science, ISSN 0882-3014, Pages 263-299.
- ADDIS T. R. AND TOWNSEND ADDIS J. J. (1995) Diagrams for Design: A schema Interpreter for Knowledge Systems Proceedings of the 15th Annual Conference of the British Computer Specialists Group on Expert Systems (ES95), pp231 - 247.
- BILLINGE D (2001) An Analysis of the Communicability of Musical Predication PhD Thesis University of Portsmouth
- BILLINGE D. & ADDIS T. R., (2002), 'Modelling the Role of Metaphor in Artistic Description', ECAI, 15th European Conference on Artificial Intelligence, Lyon, France, Workshop 17, Creative Systems: Approaches to Creativity in AI and Cognitive Science. pp 55-58.
- BILLINGE D. & ADDIS T. R., (2001), 'Some Fundamental Limits of Automated Artistic Decision Making', Symposium on AI and Creativity in Arts and Science, AISB'01, The 2001 Convention of the Society for the Study of Artificial Intelligence and the Simulation of Behaviour, 21st -24th March, University of York, UK
- COOKE, DERYCK (1959) The Language of Music. Oxford University Press
- FAUCONNIER G. & TURNER M. The Way We Think: Conceptual Blending and the Mind's Hidden Complexities. Basic Books (2002)
- GOODING, D. C. & ADDIS, T. R. (1999), 'A Simulation of model-based reasoning about disparate phenomena', Magnani, L., Nersessian, N. and Thagard, P., eds., Model Based Reasoning in Scientific Discovery, London and New York: Kluwer/Plenum. ISBN 0-306-46292-3, pp 103 - 123.
- LAKOFF, G AND JOHNSON, M (1980) Metaphors we live by. University of Chicago Press
- MEYER, L. B. (1956) Emotion and meaning in music. The University of Chicago Press
- MEYER, L. B. (1967) Music, The Arts and Ideas. University of California Press
- MEYER, L. B.(1973) Exploring Music. University of California Press

Optimality Principles for Conceptual Blending: A First Computational Approach

Francisco C. Pereira and Amílcar Cardoso

*Centro de Informática e Sistemas da Universidade de Coimbra (CISUC)

Departamento de Engenharia Informática, Polo II

Pinhal de Marrocos, Coimbra, Portugal

camara@dei.uc.pt, amilcar@dei.uc.pt

Abstract

We propose an implementation of the eight Optimality Principles from the framework of Conceptual Blending, as presented by Fauconnier and Turner (1998). Conceptual Blending explains several cognitive phenomena in the light of the integration of knowledge from different *mental spaces* onto a single *mental space*: the Blend. The Optimality Principles express general pressures that compete in the generation of the Blend. The work we present now corresponds to the Constraints module of our computational model of Conceptual Blending, also described in other papers.

1 Introduction

One big challenge for Computational Creativity is the generation of new concepts and a very interesting source of inspiration for approaching this issue comes from the framework of Conceptual Blending, from Fauconnier and Turner (1998). The Conceptual Blending (CB) framework establishes cognitive processes and principles that work in the integrations of knowledge needed in reasoning. We are developing a model of computational creativity that takes an approach to CB as a fundamental pillar. This system has been described in earlier papers (Pereira and Cardoso (2001), Pereira and Cardoso (2003)) as well as its first experiments (Pereira and Cardoso (2002)), being evident the need for developing the Optimality Principles of Conceptual Blending as an urgent improvement. These are the general guidelines that drive the process of blending and allow the differentiation between a "good" and a "bad" blend. Here, we propose a formal realization for each of the eight principles.

In the first sections, we give an overview of Conceptual Blending, so as to provide the reader the motivation and background for our approaches to the Optimality Principles. We made some exploratory experiments which are informally reported in the end of the discussion of each principle. We finalize this paper with reflections regarding the creative aspects of this work.

2 Conceptual Blending

Conceptual Blending (CB) was initially proposed by Fauconnier and Turner (1998) as part of a major framework concerning cognition and language and had the role of

explaining the integration of knowledge coming from distinct sources onto a single, independent and coherent unit, the Blend. A blend is a concept or web of concepts whose existence and identity, although attached to the pieces of knowledge that participated in its generation (the inputs), conquers gradual independence through time and use.

We find examples of blends in many sorts of situations. A blend can be an effective way to get attention and curiosity towards advertising a product (e.g. Sony's AIBO robot uses all sorts of Sony products behaving as if it were a real human) or spreading a message (e.g. the Marlboro cowboy with impotence problems). People have been making blends with creatures from the times of Greek mythology (e.g. *pegasus*) till today (e.g. *the pokemons*), natural language discourse (e.g. "John digested the book", "Sue sneezed the napkin off the table"), poetry (see Freeman (1999)). It is also claimed to be important in *scientific creativity* (Lakoff and Nunez (2000)) and in the elaboration of *cognitive instruments* (Hutchins (2002)). Many more examples and situations could be listed and studied in detail, demonstrating the ubiquity of CB.

It is noticeable that Conceptual Blending and its research community are growing, and possibly still in its early stages. It is an elegant proposal for a creative process and its relationships with language and cognition, but it carries formal vagueness across its several aspects, making it difficult even to be considered as a theory, in a *Popperian* sense. Indeed, if the big hole between the general model and the specific examples does not show any incoherence, it also does not allow falsifiability, leaving very much undefined the boundary between what is and what is not a blend. These criticisms intend to motivate work that, from our point of view, is fundamental. And support our own motivation for the present project, which is that of contributing with a formal model and implementation

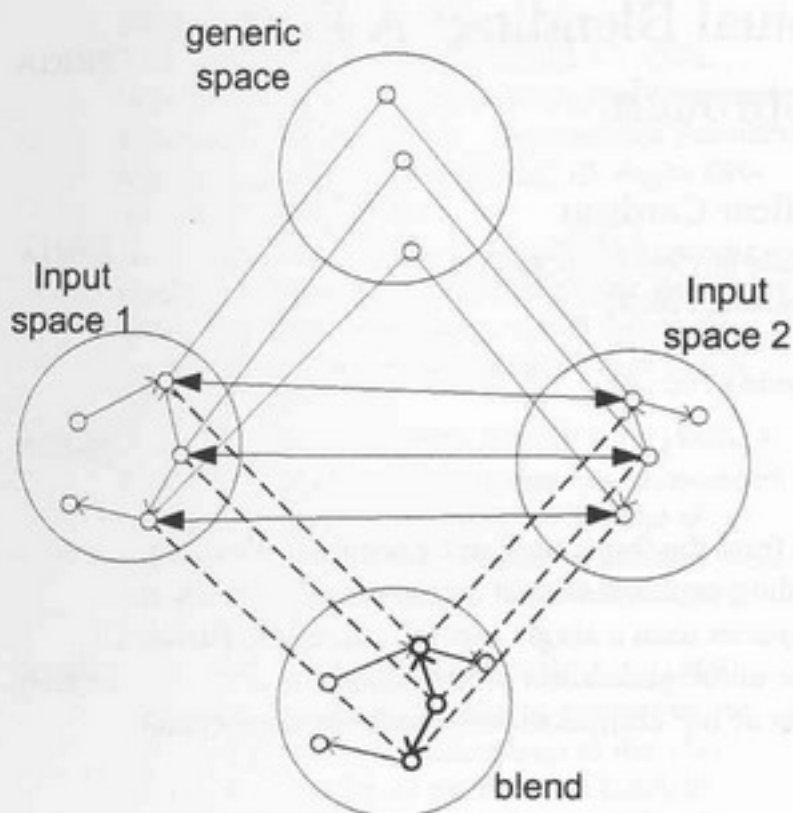


Figure 1: The Conceptual Blending Model

based on Conceptual Blending. We hope it can be useful in shedding some light on the referred issues.

In order to understand the CB framework, we must introduce a fundamental concept: the *mental space*. According to Fauconnier and Turner (1998), mental spaces are partial structures that proliferate when we think and talk, allowing a fine-grained partitioning of our discourse and knowledge structures. For simplifying, let us consider a mental space as a partial selection of knowledge from a domain, a memory of a situation, an imagined scenario or entity, essentially a knowledge structure of inter-related concepts that is explicitly or implicitly necessary for a reasoning. As AI researchers, we see a mental space representable as a *semantic network*, a *frame*, a *case* or any other symbolic knowledge structure that gathers a set of inter-related concepts towards a specific situation. In its *canonical form*, Conceptual Blending is described as involving two input mental spaces that, recurring to a cross-space mapping between them and a generic space that has general knowledge relevant for both the input domains, will generate a third one, called Blend. This new domain will maintain partial structure from the input domains and have emergent structure of its own. The process of generation of a blend can be summarized according to three general steps (Fauconnier (1997)): Composition, where new relations become available that did not exist in the separate inputs; Completion, when generic knowledge is projected into the blend, to "complete" the emergent structure; and Elaboration, in which cognitive work is performed in the blend, according to its own

emergent logic. The order of these steps may be changed and several iterations of the process may be necessary. There is a set of *governing principles*, the *Optimality Pressures*, that should drive the process of generating a "good blend" (Fauconnier and Turner (2002)):

- **Integration** - The blend must constitute a tightly integrated scene that can be manipulated as a unit. More generally, every space in the blend structure should have integration.
- **Pattern Completion** - Other things being equal, complete elements in the blend by using existing integrated patterns as additional inputs. Other things being equal, use a completing frame that has relations that can be the compressed versions of the important outer-space vital relations between the inputs.
- **Topology** - For any input space and any element in that space projected into the blend, it is optimal for the relations of the element in the blend to match the relations of its counterpart.
- **Maximization of Vital Relations** - Other things being equal, maximize the vital relations in the network. In particular, maximize the vital relations in the blended space and reflect them in outer-space vital relations. Turner and Fauconnier identify 15 different vital relations: change, identity, time, space, cause-effect, part-whole, representation, role, analogy, disanalogy, property, similarity, category, intentionality and uniqueness.
- **Intensification of Vital Relations** - Other things being equal, intensify vital relations.
- **Web** - Manipulating the blend as a unit must maintain the web of appropriate connections to the input spaces easily and without additional surveillance or computation.
- **Unpacking** - The blend alone must enable the understander to unpack the blend to reconstruct the inputs, the cross-space mapping, the generic space, and the network of connections between all these spaces
- **Relevance** - Other things being equal, an element in the blend should have relevance, including relevance for establishing links to other spaces and for running the blend. Conversely, an outer-space relation between the inputs that is important for the purpose of the network should have a corresponding compression in the blend.

As far as we know, there is no work yet towards an objective study of the optimality pressures, measuring examples of blends or specifying these principles in detail. This, we believe, disturbs considerably the appreciation and application of Conceptual Blending in scientific research, making a particular motivation for

this work being that of testing and specifying a formal proposal for the optimality pressures.

3 Basic Notions from our model

We present some basic notions that are necessary to understand this document. We use a specific type of mental space, that is static, modeless and non attached to discourse, which is closer to the notion of *domain knowledge* in AI. We call it a *domain*, comprising a theory and a set of instances. In this paper, we consider only the theory, which is represented by a *concept map* that explains the structural and causal organization of the domain. A concept map is a semantic network, with binary directed relations between the concepts¹. In CB, there is a set of *vital relations* that take a special role in the Blending process. We also consider these relations, but allow the choice of a different set of relations. In principle, these vital relations can be the source for establishing *mappings* between the input spaces, fundamental for the projection operation (two objects mapped to each other can be projected to the same concept, e.g. horse and bird projected to pegasus). Each mapping projection consists on a ternary relation $m/3$. For example, a mapping algorithm based on "property" could try to map pairs of concepts that make a valid pair of object/property (e.g. "m(property, dark, bird)"), a different one based on analogy could link pairs of analogical counterparts (e.g. "m(analogy, leg, wing)"). As far as our research goes, we are only applying an analogical mapping algorithm that finds a 1-to-1 structure mapping between the concept maps of the two domains. Knowing this is a strong limitation, we hope to address other mapping algorithms in future developments. Currently, for the sake of validation and experimentation of the system, we allow *user-defined* mappings, so as to allow conclusions independent from the mapping choice.

Finally, another important notion is that of *frames*. The reasoning behind a frame lays on the idea that concepts within it should be tightly integrated according to a situation, structure, cause-effect or any other relation that ties a set of concepts onto one, more abstract or broad, composite concept. We envisage different kinds of frames, in terms of level of abstraction. The more specific ones correspond to concepts easily identifiable as familiar objects or situation (which we can see as a kind of *prototypes*). For example, the (very much simplified) frame of "transport means" corresponds to a set of concepts and relations that, when connected together, represent something that has a container and a subpart (e.g. an engine) that serves

for locomotion.

$$\begin{aligned} \text{frame}(\text{transport_means}(X)) : \\ \text{carrier}(X, \text{people}) \leftarrow \begin{aligned} &\text{have}(X, \text{container}) \wedge \\ &\text{have}(X, Y) \wedge \\ &\text{purpose}(Y, \text{locomotion}) \wedge \\ &\text{drive}(_, X) \end{aligned} \end{aligned}$$

The more abstract frames can consist on top-level decisions or directives that decide the underlying philosophy of the blend construction. For example, if a blend satisfies the "aframe" frame, it means it maintains the structure of the input domain 1. If it satisfies the "bprojection" frame, then the concepts of input space 2 all get projected unchanged to the blend (e.g. "bird" is projected to "bird", "wing" is projected to "wing"). If a blend satisfies "aframe" and "bprojection" simultaneously, then it should² have the concepts from input space 2 organized according to the structure of input space 1. It is more complex to design these more abstract frames, thus we allow the use of programming (in *prolog* language, inside curly brackets) within a frame specification. Below we can see the programming of the "aframe".

$$\begin{aligned} \text{frame}(\text{aframe}(A)) : \\ \text{aframe}(A, \text{Blend}) \leftarrow \\ \{ \text{stats}(\text{domain1}, A), \text{current_blend}(Bl) \}, \wedge \\ \{ \text{findall}(R/X/Y, (\text{rel}(A, XA, R, YA), \text{pro-} \\ \text{jection}(Bl, A, XA, X), \text{projection}(Bl, A, \\ YA, Y)), L) \} \wedge \text{op}(\text{exists}(L)) \end{aligned}$$

Basically, "aframe" searches for all relations of domain 1 (A) and obtains a list (L) containing their projections to the blend. The $\text{op}(\text{exists}(L))$ condition is an *operator* that, when, interpreted by the frame processor, expands L into a set of concept map relations (like those in the "transport_means" frame). For "aframe" to be totally satisfied, it is necessary that all relations from input space 1 also exist in the blend.

There are also frames of intermediate level of abstraction, which aren't either as specific as "transport_means" or as abstract as "aframe". For example, "new_creature" is concerned with finding a "creature" (thus having its specific properties - e.g. being a "living being") that didn't exist before in either domains or existed but not as a "creature" (e.g. a "flying snout").

When we say that a frame f is satisfied in the blend b , we mean all its premises are true in the domain b . We see frames as *information molds* and building a blend for a given situation should depend much on the choice of these structures, either being structures towards which the blend self-organizes or as pragmatic goals or query specifiers that the blend is expected to accomplish.

¹First order logic predicates with arity 2, e.g. $\text{isa}(\text{bird}, \text{aves})$; $\text{purpose}(\text{wing}, \text{fly})$

²Depending on the mappings, the concepts considered in aframe and bprojection may become separate in the inputs (and so there wouldn't be any systematic relation between concepts from input spaces 1 and 2), yet this would receive little value in the measures presented in this paper.

4 Optimality Principles

Following the F&T notion of Optimality Principles, the pressures that should lead towards stable, integrated new blends, we propose now a set of measures that should reflect as much as possible the rationale behind each principle. In order to give a clearer idea of its individual effect in our blending system, we present a brief report of experiments we made. These experiments consisted in running a parallel search method, a genetic algorithm¹, to retrieve blends from the search space. The input domains were the domains of *horse* and *bird* (see tables 1 and 2), meaning that the expected results range from the unchanged copy of one (or both) of the concepts to a horse-bird (or bird-horse) which is a combination of selected features from the input domains. The construction of these domains was subject to the following constraints: they should be concept networks in which nodes are concepts and arcs are relations; the concepts should be connected to the ontology in the generic space through an "isa" relation; the relations used should be present in the Generalized Upper Model (GUM) hierarchies (J. Bateman and Fabris (1995)) or be subtypes of them. GUM is a general top-level ontology that has two hierarchies (elements and relations) that comprise abstract relations, properties, spatial relationships, among others. Although allowing a normalization of the concept maps, the constraints in the construction of the domains don't avoid, *per se*, the biasing or ingenious tailoring. For this reason, in this paper and in the exploratory experiments we show, we don't give special attention to a qualitative *reading* of the results or use them to demonstrate its validity, instead we are interested in reporting the effects each measure has on the results and on the search landscape.

The generic domain (in tables 3 and 4) consists on a simple general ontology, a set of frames and integrity constraints. We applied 3 different mappings (figure 2), all generated by the Mapper module. These mappings range from being very small (only four mapping correspondences) to large (21 mapping correspondences), from non-surprising associations (e.g. "animal" and "animal") to nonsense (e.g. snout and lung). For each mapping, we tested the seven optimality pressures. Each of these comprising 30 runs³.

4.1 Integration

Frames have a natural integration role because they gather knowledge around abstractions, tightening the links between concepts. Assuming the set F of frames that are satisfied in a blend, we define the *frame coverage* of a domain to be the set of relations from its concept map that belong to the set of conditions of one or more frames in F . The larger the frame coverage of the blend, the more it is integrated. Yet, a blend that is covered by

many frames should be less integrated than a frame with the same coverage, but with less frames. In other words, if a single frame covers all the relations of a blend, it should be valued with the maximal integration, whereas if it has different frames being satisfied and covering different sets of relations, it should be considered less integrated. The intuition behind this is that the unity around an integrating concept (the frame) reflects the unity of the domain. The Integration measure we propose varies according to this idea. It also takes integrity constraints into account so that, when a frame violates such a constraint, it is subject to penalty.

Definition 4.1 For a frame f with a set C of conditions $B_i \cup \neg B_i$, a blend b , with a concept map CM_b , a blendoid⁴ CM_{B+} , the concept map of the blendoid and VI , the set of integrity constraints⁵ that are violated in the frame, the integration value, I_f is defined by:

$$I_f = \left(\frac{\#C}{\#CM_b} \times (1 - \iota)^{\#VI} \right) \times \left(1 + \frac{\#CM_b}{\#CM_{B+}} \right) / 2$$

being ι an penalty factor between 0 and 1, a value that penalizes a frame for each violation to integrity constraints. An integrity constraint is violated if its premises are true. In the context of the integration measure of frame f above, f violates integrity i if the conditions Ci of i are verified and $Ci \cap C \neq \emptyset$. In other words, f needs to violate i in order to be integrated.

We would like to clarify the above formula further more: the first factor represents the ratio of coverage of b w.r.t. f ; the second factor means that each integrity constraint violation implies an exponential discount; the third factor serves the purpose of maximizing the size of the blend (if two frames have the same ratio of coverage, the one that contains more relations should have higher integration); the division by 2 aims to normalize the result between 0 and 1.

While the value for a single frame integration is described above, the integration measure of a domain w.r.t. a set of frames is not necessarily straightforward. At a first sight, it is appealing to just sum the values of integration of all frames, or of the union of them. Or even their intersection. But this would lead to wrong results, because a set of frames could not be reduced to a single frame from the point of view of integration. In this measure, we want to stimulate unity, coverage and take into account the strength of each frame individually. In terms of unity, we argue that the set of relations that make the "core" of all the frames that are satisfied, i.e. the intersection of the sets C of conditions of all frames, should be highly valued. On the other side, the coverage of this "core" will be smaller than the overall coverage (or equal, if

³A run is an entire evolutive cycle, from the initial population to the population in which the algorithm stopped

⁴The concept map that contains all the possible relations the blend may have

⁵Rules with *false* conclusion as in table 3

the frames have equivalent C sets), which leads us to take into account the disjoint sets of relations of the frames. Finally, the integration of each individual frame (as defined above) should also be present in the overall measure. These last two issues (the overall coverage and the integration of individual frames) are subject to a disintegration factor because they reflect the existent of different, not totally intersected, frames. We propose this factor, α , to be a configurable value from the interval $[1, 0]$. It is now time to present our proposal for the *IntegrationMeasure* of a blend:

Definition 4.2 For a set of frames $f_i \in F_b$, with F_b being the set of the frames that have their conditions (C_i) satisfied in the blend b

$$Integration = I_{\bigcap_0^i C_i} + \alpha \times Uncoverage \times \sum_0^i I_{f_i}$$

The *Uncoverage* value consists on the ratio of relations that do not belong to the intersection of all frames w.r.t. the total number of relations considered in the frames:

$$Uncoverage = \frac{\# \bigcup_0^i C_i - \# \bigcap_0^i C_i}{\# \bigcup_0^i C_i}$$

We think the integration measure is a fundamental brick of the blending process. It leads the choice of the blend to something *recognizable* as a whole, fitting patterns that help to determine and understand what a *new* concept is.

Experiments: An immediate conclusion about the effect of Integration is that frames behave as *attractor* points in the search space. Moreover, the frames with a larger coverage tend to be preferred, although when too large (like *aprojection* or *aframe*) they are dropped away. The evolution is directed to a compromise of coverage and satisfiability. More specifically, when it satisfies a frame like *pw_based_explanation*, the resulting Integration value is a local maximum or a point in its neighborhood (because sometimes other, related, frames were also found) and "jumping" to another area of the search space becomes difficult.

Another conclusion to take concerns to the observation that the complexity of the search space landscape grows with mapping size. In fact, when we have a mapping of size 2, the algorithm only finds two different solutions and the better rated (possibly a global maximum) is achieved in 77% of the runs, but with a mapping of size 5, it returns six different blends, being the best choice retrieved only 43% of the times. To confirm this conclusion, the mapping of size 21 lead the algorithm to 16 different maxima, being the best one found only 7% of the times. A good compensation for this apparent loss of control is that the returned values are clearly higher (0.68, for the best) than in the small mappings (0.22), meaning that with big mappings there are many more possibilities to find integrated

blends.

The resulting concept maps consist on exactly the relations that are covered by the satisfied frame or combination of them, more specifically there were two frames that were very persistent: *pw_based_explanation* and *purposeful_subpart*. In some blends, the former was present multiple times (e.g. a part-whole explanation of a horse, with part-whole explanations of its subparts) and both were plenty of times combined.

4.2 Topology

The Topology optimality pressure brings *inertia* to the blending process. It is the constraint that drives against change in the concepts because, in order to maintain the same topological configuration as in the inputs, the blend should maintain exactly the same neighborhood relationships between every concept, ending up being a projected copy of the inputs. In *real blends*, this pressure is normally one that is disrespected without big loss in the value of the blend. This is due to the *imagination* context that normally involves blends, i.e., novel associations are more tolerable.

In our Topology measure, we follow the principle that, if a pair of concepts, x and y , is associated in the blend by a relation r , then the same relation must exist in the inputs between the elements from which x and y were projected. We say that $r(x, y)$ is *topologically correct*. Thus, the value of Topology corresponds to the ratio of topologically correct relations in the concept map of the blend.

Definition 4.3 For a set $TC \subseteq CM_b$ of topologically correct relations, defined as

$$TC = \{r(x, y) : r(x, y) \in CM_1 \cup CM_2\}$$

where CM_1 and CM_2 correspond to the concept maps of inputs 1 and 2, respectively⁶. The topology measure is calculated by the ratio:

$$Topology = \frac{\#TC}{\#CM_b}$$

Intuitively, this measure represents the amount of relations from the inputs that got projected onto the topologically equivalent position in the blend. At the moment, the only way to violate topology is by having a pair of concepts projected to the same one (e.g. "horse" and "bird" projected to "horse"), bringing a new relation exclusive to one the domains (e.g. ability(bird, fly) projects to ability(horse, fly)). Topology thus decreases as fusion or transfer projections are made.

Experiments: In all the experiments with Topology, the final results were valued 100%, meaning that this constraint is easily fully accomplished, independently of the

⁶In other words, TC is the intersection of the concepts maps of the blend and the input spaces

mapping. An interesting fact is that there is a multitude of solutions in the *search landscape* of Topology, showed by the amount of different final results in each mapping. Intuitively, and observing the short duration of each run, this means that, wherever the search starts, there is always a Topology optimal point in the neighborhood. Topology is more an inertia than a transformation force because it values knowledge that remains the same. In our horse-birds, this pressure projects wings, beaks and claws (i.e. concepts from the bird domain) to the blend but isolates them unless there is strong evidence to connect to horses, legs and snouts (i.e. concepts from the horse domain).

4.3 Pattern Completion

The Pattern Completion pressure brings the influence of patterns being them present in the *inputs* or come from the *generic space*. Sometimes a concept (or a set of concepts) may seem incomplete but making sense when "matched against" a pattern.

At present, in the context of this work, a pattern is described by a frame, i.e. we don't distinguish these two concepts, and therefore pattern completion is basically frame completion. Here, as in the definition of this principle, the completing knowledge becomes available from "outside", not as a result of projection. This means the act of completing a frame consists on asserting the truth of the ungrounded premises, a process that happens only after a sufficient number of premises is true. We call this the *evidence threshold*. An interesting approach in Pattern Completion we would like to consider is that of *abduction*. I.e., if the conclusions are satisfied, why not derive the premises?

The evidence threshold e of a frame f_i with regard to a blend b is calculated according to the following.

$$e(f_i, b) = \frac{\#Sat_i}{\#C_i} \times (1 - \iota)^{\#VI}$$

where Sat_i contains the conditions of each f_i that are satisfied in b , ι is the integrity constraint violation factor and VI the set of violated integrity constraints.

As in the integration pressure, we have the problem of taking into account multiple frames. This time, given that we are evaluating possible completion of subsets of relations, instead of sets of relations that are actually verified in the domain, it is difficult to find such a linear rationale (e.g. would two patterns each with individual completion x value higher than three having each slightly less than x ?). As a result, we propose to find the union of the patterns and then estimate its own evidence threshold:

Definition 4.4 The Pattern Completion measure of a blend b with regard to a set F with n frames is calculated by

$$PatternCompletion = e(\cup_0^n f_i, b)$$

This measure has a very important role in increasing the potential of the blend, for it brings the "seeds" that may be used in the Completion and Elaboration phases.

Experiments: The first conclusion to take from the experiments with Pattern Completion is that the size of the resulting concept maps tend to grow as the evolution progresses, although there is no linear correlation found. This has a simple interpretation, given that this measure stimulates the appearance of patterns (frames) that are only partially completed. In doing so, it drives the blend to partially complete (i.e., instantiate partially its conditions) the highest possible number of frames, leading, in each case, to several sets of relations that fit into those frames without satisfying them (e.g. wings are projected, they serve to fly, but they are not attached to anything).

In which respects to the *search landscape*, it seems to be very rich in local maxima. This fact is not unexpected, considering the discussion of the previous paragraphs, the number of different frames available (9) and all their different combinations. The most constant results came from mapping 2, with 13% of the best result obtained and 20% of the second best. An interesting remark is that the resulting local maxima always fall within a very strict range of values (of maximum amplitude 0.11).

4.4 Maximization of Vital Relations

For the maximization of vital relations, we estimate the impact of the vital relations to the blend calculated by the ratio of vital relations w.r.t. the whole set of possible vital relations the blendoid. The blendoid is the largest possible blend that can be obtained from a given mapping and is calculated by projecting to it every concept (i.e., there is no selective projection) regardless of integrity constraint violation or any other constraint. Since it has the largest set of potential relations, it also has the maximum possible of vital relations.

Definition 4.5 Let Υ be a set of vital relations. From the concept map of the blend b , we may obtain the set of vital relations in b , BVR :

$$BVR = \{r(x, y) : r(x, y) \in CM_b \wedge r \in \Upsilon\}$$

From the blendoid (the largest possible blend), B^+ , we have BVR^+ :

$$BVR^+ = \{r(x, y) : r(x, y) \in CM_B^+ \wedge r \in \Upsilon\}$$

Finally, the Maximization of Vital Relations measure is calculated by the ratio

$$Maximization_VR = \frac{\#BVR}{\#BVR^+}$$

Experiments: The influence of Maximization of Vital Relations in the results is straightforward, given that its highest value (1) reflects the presence, in the blend, of all the

vital relations that exist in the inputs, independently of the projections of the concepts or non-vital relations (which become *noise* in the sense that these appear randomly and making no difference to the value of the measure, yet confusing the "reading" of the concept map). As the evolution goes on in each run, the value grows until reaching the maximum reasonably early. For each set of 30 runs, it reached the value 1 a minimum of 93% of the times, and the remaining achieved at least a value of 0.95. As in Topology, the search space of Maximization of Vital Relations is very *simple* since there is a global maximum in the neighborhood of (almost) every point. On the other hand, since there is control on *noise*, the resulting concept maps show unity only in the subset of vital relations.

4.5 Intensification of Vital Relations

Intensification of Vital Relations is the principle that maximizes the concentration around a specific vital relation. I.e., while the Maximization of Vital Relations favors the creation in the blend of vital relations in general as opposed to "regular" relations, Intensification is based on focussing a specific vital relation. The former relates "new" vital relations with "new" relations in the blend; the latter relates vital relations with themselves. Thus, we need a notion of "intensity" of a vital relation. For such, we argue that a vital relation is considered more "intense" when there is more evidence of its strength. This evidence should be dependent on the kind of vital relation we are dealing with. For example, an "analogy" vital relation between two concepts is stronger when there is also a systematical association between the neighborhood concepts (the systematicity principle). In fact, systematicity is the only "intensity" heuristic we have now and its calculation ($Int_{analogy}$) is straightforward. For a mapping (of size n)

$$Int_{analogy} = \frac{\#analogy_transfers}{n}$$

where an *analogical transfer* consists on a projection of a concept x from input space 1 to y in the blend (where y is the analogical counterpart of x in the input space 2). Considering several different "intensity" heuristics, the evaluation of this pressure takes the point of view that a blend that applies only one vital relation, with intensity x , should have higher measure than a blend with n vital relations, each with intensity x/n (the sum would thus be x). So we want to favor "concentration" of vital relations.

Definition 4.6 Let $v_i \in \Upsilon$ be a vital relation and the set VRB_{v_i} , of the instances of vital relation v_i in the blend, defined by $VRB_{v_i} = \{v_i(x, y) : v_i(x, y) \in CM_b\}$. Assuming a value Int_{v_i} of intensity of the vital relation v_i , the measure of Intensification of Vital Relations is calculated by:

$$Intensification_VR = \frac{\sum_0^n Int_{v_i}^2}{(\sum_0^n Int_{v_i})^2}, n = \#\Upsilon$$

Intensification is thus higher when there is more "concentration" (e.g. $Int_{v_1} = 2, Int_{v_2} = 2 \Rightarrow Intensification = 8/16$; $Int_{v_1} = 4 \Rightarrow Intensification = 1$).

Experiments: The behavior of Intensification of Vital Relations is similar to that of Maximization in the sense that the search landscape is not very complex. From wherever the search starts, there is a high probability (of at least 70%, in the worst case) of finding the maximum value in the neighborhood. It is important to point, though, that we are only applying one heuristic (analogical transfer), and so the results couldn't be different. Given this fact, we must say that we cannot claim or discuss much about this Intensification of V.R. proposal unless we find other heuristics and mapping procedures other than analogy.

4.6 Unpacking

Unpacking is the ability to reconstruct the whole process starting from the blend. In our view, such achievement underlies the ability to reconstruct the input spaces, specifically. I.e., the reconstruction of the input spaces from the blend demands the assessment of the cross-space mappings, the generic space and other connections. Thus, what we are proposing is that Unpacking can be reduced to the ability to reconstruct the inputs. This is so because there is no way to properly reconstruct the inputs without a reconstruction of the cross-space mappings, generic space and the connections between spaces.

Unpacking should take the point of view of the "blend reader", i.e., someone or something that is not aware of the process of generation, thus not having access to the actual projections. Being such, this "reader" will look for patterns that point to the "original" concepts. Once again we use the idea of *frames*, more specifically the *defining frame* of a concept, which comprises its immediately surrounding concepts and relations. For example, if the concept "wing" were projected onto x in blend, the defining frame with regard to the "bird" domain would consist on $purpose(x, fly)$, $conditional(x, fly)$, $quantity(x, 2)$ and $pw(x, bird)$. The more of these relations are found in the blend, the more likely it is that the "reader" will find easy to understand the relationship between x and "wing".

Definition 4.7 Given a blend b and an input space d , the concept x (which is the projection of the element x^d of input space d to the b), has a defining frame $f_{x,d}$ in d consisting on

$$f_{x,d} = C_0, C_1 \dots C_n \rightarrow true$$

where $C_i \in \{r(x, y) : r(x^d, y) \in CM_d\}$.

Assuming that k is the number of conditions (C_i) of $f_{x,d}$ that are satisfied in the blend, the unpacking value of x with regard to d (represented as $\xi(x, d)$) is

$$\xi(x, d) = \frac{k}{n}$$

We calculate the *total estimated unpacking value* of x as being the average of the unpacking values with regard to the input spaces. Thus, having input spaces 1 and 2, we have

$$\xi(x) = \frac{\xi(x, 1) + \xi(x, 2)}{2}$$

Definition 4.8 Let $\mathcal{X}$ be the set of n concepts of the blend b , generated from input spaces 1 and 2. The Unpacking value of b is calculated by

$$\text{Unpacking} = \frac{\sum_{i=0}^n \xi(x_i)}{n}, x_i \in \mathcal{X}$$

Experiments: The results of the Unpacking measure show that it has a notorious side effect on the size of the blend, it drives it to very small sets (between 0 and 5) of relations. The interpretation here is straightforward: the ratio of *unpackable* concepts is highly penalized in bigger sets because of the projected relations that come as side effect of the projection of (*unpackable* or not) concepts. These relations *confuse* the unpacking algorithm so that it leads the evolution to gradually select the smaller results.

The maxima points also correspond to the value 1, but it seems, from the experiments, that there is a very limited set of such individuals, achieved in the majority (at least 93% for each mapping) of the experiments.

4.7 Web

The Web principle concerns to being able to "run" the blend without cutting the connections to the inputs. In our opinion, this is not an independent principle, being co-related to those of Topology and Unpacking because the former brings a straightforward way to "maintain the web of appropriate connections to the input spaces easily and without additional surveillance or computation" and the latter measures exactly the work needed to reconstruct the inputs from the blend. It is not to say that Web is the same as Topology or Unpacking, what we are arguing is that, on one side, Topology provides a pressure to maintain the most fundamental connection to the input: the same structure; on the other side, Unpacking evaluates the easiness of reestablishing the links to the inputs. These two values combined in a weighted sum yield, we propose, an estimation of the strength of the web of connections to the inputs:

$$\text{Web} = \alpha \times \text{Topology} + \beta \times \text{Unpacking}$$

with $\alpha + \beta = 1$. Since this is not an independent variable, making independent experiments with the Web measure would not add any valuable conclusion. In a subsequent publication, we plan to focus on correlation of measures, where we may explore the behavior of this measure.

4.8 Relevance

The notion of "relevance" or "good reason" for a blend is tied to the context and goal of the blending generation.

A blend, or a part of it, may be more or less relevant dependent of what it is for. Once again, frames take a fundamental role as being "context" specifiers, (i.e., the set of constraints within a frame describe the context upon which the frame is fulfilled). Therefore, having a set of goal frames, which could be selected from any of the existent domains or specified externally, a blend gets the maximum Relevance value if it is able to satisfy all of them.

An aspect of the goal frames is that they allow the application of queries. For example, if we want to find a concept that "flies", we could build a goal frame with the relation *ability(x, fly)*. The blends that satisfy this frame would have high relevance.

Definition 4.9 Assuming a set of goal frames, F_g , the set F_b of the satisfied frames of blend b and the value PCN_F for the pattern completion of a set of frames F in blend b , as described in section 4.3, we have

$$\text{Relevance} = \frac{\#(F_g \cap F_b) + \#F_u \times PCN_{F_u}}{\#F_g}$$

where F_u , the set of unsatisfied goal frames, consists on $F_u = F_g - F_b$.

Experiments: The first part of the test on Relevance focussed on making a single relation query. In this case, we asked for "something that flies" (*ability(., fly)*). The results were straightforward in any mapping, accomplishing the maximum value (1) in 100% of the runs, although the resulting concept maps did not reveal necessarily any overall *quality* or unity. In other words, the evolution took only two steps: when no individual has a relation "ability(., fly)", therefore with value 0; when a relation "ability(., fly)" is found, yielding a value 1, independently of the rest of the concept map.

The second part of the test on Relevance, by adding a frame (*ability_explanation*) to the query, revealed similar conclusions. There was no sufficient knowledge in any of the input domains to satisfy this new frame completely, so the algorithm searched for the maximum satisfaction and reached it 100% of times in every mapping. So the *landscape* seems to have one single global and no local maxima, reflecting the integration of the two parts of the query. If there were separate frames, it is expectable the existence of local maxima. Intuitively, the *search landscapes* of Integration and Relevance seem to be similar.

5 Discussion and further work

A first conclusion we took from this work says that the eight optimality principles can be reduced to seven (since Web is not independent). Even more, given the power of language we use in frames, some of the principles can be *coded* within a frame, namely Topology and Unpacking, and accomplished via a query measured by

Relevance. This reduces our number to five. Yet, we don't know whether this reduction reflects a lack in the CB framework or in our interpretation of it.

We have already stated that the main motivation of our system is to generate new concepts out of previous knowledge. An interesting study to follow, and which has already been made (Pereira and Cardoso (2003)), concerns the necessary parameters that lead to specific concepts (e.g. a pegasus) and how do the changing on these bring different concepts (other creatures). Such experiments provide a bigger insight on the creative potential of our system. Perhaps a more important study, in the sense that it brings a validation to our model, is the application of literature examples from CB (Coulson (2000)) and Conceptual Combination (Costello and Keane (2000)), observing the effects of the application of optimality constraints presented in this paper as well as frames.

6 Acknowledgements

We would like to thank the reviewers for their constructive comments and to Pablo Gervás and Margaret Freeman for participating in discussions that gave rise to some ideas in this paper.

References

- Fintan J. Costello and Mark T. Keane. Efficient creativity: Constraint-guided conceptual combination. *Cognitive Science*, 2000.
- Seana Coulson. *Semantic Leaps: Frame-shifting and Conceptual Blending in Meaning Construction*. New York and Cambridge: Cambridge University Press, 2000.
- Gilles Fauconnier. *Mappings in Thought and Language*. Cambridge University Press, 1997.
- Gilles Fauconnier and Mark Turner. Conceptual integration networks. *Cognitive Science*, 22(2):133–187, 1998.
- Gilles Fauconnier and Mark Turner. *The Way We Think*. Basic Books, 2002.
- Margaret Freeman. The role of blending in an empirical study of literary analysis. In *Proceedings of the 6th International Cognitive Linguistics Conference*, 1999.
- Edwin Hutchins. Material anchors for conceptual blends. *forthcoming*, 2002.
- B. Magnini J. Bateman and G. Fabris. The generalized upper model knowledge base: Organization and use. *Towards very large knowledge bases: knowledge building and knowledge sharing*, 1995.
- George Lakoff and Rafael Nunez. *Where Mathematics Comes From: How the Embodied Mind brings Mathematics into Being*. New York:Basic Books, 2000.
- Francisco C. Pereira and Amilcar Cardoso. Knowledge integration with conceptual blending. In *Proceedings of AICS 2001*, 2001.
- Francisco C. Pereira and Amilcar Cardoso. The boat-house visual blending experience. In *Proceedings of the Symposium for Creativity in Arts and Science of AISB 2002*. AISB, 2002.
- Francisco C. Pereira and Amilcar Cardoso. The horse-bird creature generation experiment. *submitted*, 2003.

		vegetable_food	↔	vegetable
		food	↔	food
ear	↔	wing		
snout	↔	bird		
eye	↔	lung		
mouth	↔	feathers		
2	↔	2		
hear	↔	fly		
1				
		run	↔	fly
		cargo	↔	pet
		neigh	↔	chirp
		snout	↔	lung
		mane	↔	feathers
		tail	↔	beak
		leg	↔	eye
mouth	↔	beak		
snout	↔	bird		
eye	↔	lung		
ear	↔	feathers		
eat	↔	eat		
2				
		hoof	↔	wing
		4	↔	2
		eye	↔	leg
		ear	↔	claw
		hear	↔	catch
		grass	↔	grass
		3		

Figure 2: The three mappings

isa(horse, equinae)	pw(leg, horse)	purpose(horse, food)
isa(equinae, mammal)	purpose(leg, stand)	sound(horse, neigh)
existence(horse, farm)	pw(hoof, leg)	purpose(mouth, eat)
existence(horse, wilderness)	purpose(horse, traction)	purpose(ear, hear)
pw(snout, horse)	eat(horse, grass)	color(mane, dark)
pw(mane, horse)	ability(horse, run)	size(mane, long)
pw(tail, horse)	carrier(horse, human)	material(mane, hair)
quantity(hoof, 4)	quantity(leg, 4)	
pw(eye, snout)	quantity(eye, 2)	
pw(ear, snout)	quantity(ear, 2)	
pw(mouth, snout)	purpose(eye, see)	
motion_process(horse, walk)	ride(human, horse)	
taxonomicq(horse, ruminant)	purpose(horse, cargo)	

Table 1: The domain theory of *horse*

isa(bird,aves)	existence(bird, house)	isa(parrot, bird)
isa(aves,oviparous)	purpose(bird, pet)	isa(nest, container)
lay(oviparous, egg)	existence(bird, wilderness)	role_playing(bird, freedom)
purpose(bird, food)	purpose(eye, see)	ability(parrot, speak)
smaller_than(bird, human)	purpose(beak, chirp)	purpose(claw, catch)
pw(lung, bird)	motion_process(bird, fly)	purpose(wing, fly)
purpose(lung, breathe)	quantity(eye, 2)	pw(claw, leg)
isa(paradise_bird, bird)	quantity(wing, 2)	pw(beak, bird)
isa(owl, bird)	quantity(claw, 2)	pw(eye, bird)
ability(bird, fly)	pw(wing, bird)	quantity(leg, 2)
pw(feathers, bird)	conditional(wing, fly)	pw(leg, bird)
purpose(beak, eat)	sound(bird, chirp)	purpose(leg, stand)
pw(straw, nest)		

Table 2: The domain theory of *bird*

isa(entity,something)	isa(mammal, animal)	isa(house, human_setting)
isa(situation,something)	isa(animal, living_entity)	isa(setting,space_location)
isa(state,situation)	isa(oviparous, animal)	isa(space_location,spatial_entity)
isa(process,situation)	isa(behavior, property)	isa(physical_entity,spatial_entity)
isa(temporal_entity,entity)	isa(human_setting, setting)	isa(physical_object,physical_entity)
isa(spatial_entity,entity)	isa(bird, existence, wilderness)	isa(property_measure,information_entity)
isa(information_entity,entity)	isa(wilderness, setting)	isa(imaginary_spatial_entity,spatial_entity)
isa(human, primate)	isa(farm, human_setting)	isa(property,information_entity)
isa(primate, mammal)	isa(equinae, mammal)	isa(living_entity,physical_entity)
	:	

shape(X, Y),shape(X,Z), $Y \neq Z \rightarrow \text{false}$
 quantity(X, Y), quantity(X,Z), $Y \neq Z \rightarrow \text{false}$
 behavior(X, friendly), behavior(X, dangerous) $\rightarrow \text{false}$
 actor(X,...), not isaN(X, action) $\rightarrow \text{false}$
 pw(A,A) $\rightarrow \text{false}$

Table 3: The generic space concept map and integrity constraints

Frame name	Conditions
aframe	The blend contains the same relational structure of input 1
aprojection	The blend contains the same concepts of input 1
bframe	The blend contains the same relational structure of input 2
bprojection	The blend contains the same concepts of input 2
pw_based_explanation	Set of part-whole relations associated to a concept
transport_means	Features expected in a generic transport means
purposeful_subpart	Set of relations that justify the existence of a subpart of a concept
new_ability	A concept has an ability relation not existent in any of the inputs
new_creature	A concept is a living thing that did not exist (or wasn't such) in any of the inputs
new_feature	A concept has a feature relation not existent in any of the inputs
ability_explanation	A concept has an ability and it is explained by several features (<i>what does the ability, what is it for, any conditions necessary</i>)

Table 4: Frames of the generic space

Creative Abduction as Active Shaping of Knowledge Epistemic and Ethical Mediators

Lorenzo Magnani

Department of Philosophy and Computational Laboratory; University of Pavia; Pavia; Italy
Philosophy, Science, and Technology Program; Georgia Institute of Technology; Atlanta; GA; USA
lmagnani@unipv.it, lmagnani@cc.gatech.edu

Abstract

The concept of *manipulative abduction* is devoted to capturing the role of action in many interesting situations: action provides otherwise unavailable information that enables the agent to solve problems by starting and performing a suitable abductive process of generation or selection of hypotheses. Many *external representations*, even if in some cases inert from an epistemological point of view, can be transformed into what is called *epistemic mediators*, active in creative abductive reasoning. An often neglected side of human creativity is related to emotional, artistic, and ethical aspects, and concerns the active shaping of values in an esthetical and ethical world. I will present some aspects of this kind of reasoning in the case of scientific and ethical thinking; moreover, I will illustrate some aspects of what I call "ethical mediators" in their activity of shaping and reshaping ethical worth of human beings and collectives.

1 The Inexplicability of Creativity

Creativity is certainly an important aspect of our definition of "intelligence" but the literature associates many different notions to creativity. This ambiguity has brought to a lack of consensus in the research community. The common views associate to creativity unusual and mysterious qualities that drive the concept of creativity to a confused verbosity. Statements like "to break the rules", "to think different", "to destroy one *Gestalt* in favor of a better one", and "to arrange old elements into a new form", present in the field of psychological research on creativity since 1950s, certainly do not clarify the topic, and seem to lead to the Freudian conclusion that creativity cannot be understood. This conclusion has also been supported by many philosophers who studied conceptual change in science during the second half of the last century. They distinguished between a logic of discovery and a logic of justification (i.e. between the psychological side of creation and the logic argument of proving new discovered ideas by facts). The consequent conclusion was that a logic of discovery (and a *rational* model of discovery) could not exist: scientific conceptual change is cataclysmic and irrational, dramatic, incomprehensible and discontinuous. Many other studies already argued that creativity can be understood (Boden, 1991, Sternberg, 1999, Sternberg, Kaufman, and Pretz, 2002), but paid attention mainly to the psychological and experimental aspects, disregarding the philosophical, logical, and computation ones.

In AI research, however, since Simon, two characteristics seem to be associated to creativity: the *novelty*

of the product and the *unconventionality* of the process that leads to the new product. Hence, in a strictly *pragmatic* sense, when we can clarify what behavior we are looking for, we could implement it in a machine: a methodological criterion enables us to define and consider just those practical effects we conceive to be associated with novelty and unconventionality (cf. Buchanan, 2001).

I maintain we can overcome many of the difficulties of creativity studies developing a theory of abduction, in the light of Charles Sanders Peirce's first insights.

2. Abduction and Epistemic Mediators

If we decide to adopt this kind of methodology it is necessary to develop a cognitive model of creativity able to represent not only "novelty" and "unconventionality", but also some features commonly referred to as the entire creative process, such as the expert use of background knowledge and ontology (defining new concepts and searching heuristically among the old ones) and the modeling activity developed in the so called "incubation time" (generating and testing, transformations in the space of the hypotheses). The philosophical concept of *abduction* may be a candidate to solve this problem, and offers an approach to model creative processes of hypotheses generation in a completely explicit and formal way, which can fruitfully integrate the narrowness proper of a merely psychological approach, too experimentally human-oriented.

2.1 Theoretical and manipulative abduction

A hundred years ago, C. S. Peirce (CP, 1931-1958) coined the concept of abduction in order to illustrate that the process of scientific discovery is not irrational and that a methodology of discovery is possible. Peirce interpreted abduction essentially as an “inferential” *creative process* of generating a new hypothesis. Abduction has a logical form (fallacious, if we model abduction by using classical logic) distinct from deduction and induction. Reasoning which starts from reasons and looks for consequences is called *deduction*; that which starts from consequences and looks for reasons is called *abduction*.

Abduction is the process of *inferring* certain facts and/or laws and hypotheses that render some sentences plausible, that *explain* or *discover* some (eventually new) phenomenon or observation; it is the process of reasoning in which explanatory hypotheses are formed and evaluated. There are two main epistemological meanings of the word abduction (Magnani, 2001a): 1) abduction that only generates “plausible” hypotheses (“selective” or “creative”) and 2) abduction considered as inference “to the best explanation”, which also evaluates hypotheses. To illustrate from the field of medical knowledge, the discovery of a new disease and the manifestations it causes can be considered as the result of a creative abductive inference. Therefore, “creative” abduction deals with the whole field of the growth of scientific knowledge. This is irrelevant in medical *diagnosis* where instead the task is to “select” from an encyclopedia of pre-stored diagnostic entities. We can call both inferences ampliative, selective and creative, because in both cases the reasoning involved amplifies, or goes beyond, the information incorporated in the premises.

*Theoretical abduction*¹ certainly illustrates much of what is important in creative abductive reasoning, in humans and in computational programs, but fails to account for many cases of explanations occurring in science when the exploitation of environment is crucial. It fails to account for those cases in which there is a kind of “discovering through doing”, cases in which new and still unexpressed information is codified by means of manipulations of some external objects (*epistemic mediators*). The concept of *manipulative abduction*² captures a large part of scientists’ and physicians’ thinking where the role of action is central, and where the features of this action are implicit and hard to be elicited

Peirce uses the terms “inference” and “inferential process” to refer to abduction. It is useful to try to clarify the meaning of the term “inference” as considered by Peirce’s thought. Peirce stated that all thinking is in signs, and signs can be icons, indices or symbols. Moreover, all *inference* is a form of sign activity, where the word sign includes “feeling, image, conception and other representation” (CP 5.283), and, in Kantian words, all synthetic forms of cognition. Feelings, images, simulations, etc., are currently characterized as forms of model-based reasoning (Magnani & Nersessian, 2002). Consequently, following Peirce, we can say that a considerable part of thinking activity is *model-based* (cf. footnote 1), that most of the forms of constitution of phenomena are characterized in a model-based way. I use the term “model-based reasoning” following Nersessian (1995), that is, to indicate the construction and manipulation of various kinds of representations, not necessarily sentential and/or formal. Scientific concept formation, scientific discovery, and – as we will see – diagnostic reasoning are often related to *heuristic* procedures that resort to mental/internal but also to external “models” and representations.

Peirce gives an interesting example of model-based abduction (Magnani, 1999) related to sense activity: “A man can distinguish different textures of cloth by feeling: but not immediately, for he requires to move fingers over the cloth, which shows that he is obliged to compare sensations of one instant with those of another” (CP 5.221); this idea surely suggests that abductive movements also have interesting extra-theoretical characteristics and that there is a role in abductive reasoning for various kinds of manipulations of external objects (cf. below, the problem of “action-based, manipulative abduction”). One more example is given by the fact that the perception of tone arises from the activity of the mind only after having noted the rapidity of the vibrations of the sound waves, but the possibility of individuating a tone happens only after having heard several of the sound impulses and after having judged their frequency. Consequently the sensation of pitch is made possible by previous experiences and cognitions stored in memory, so that one oscillation of the air would not produce a tone.

Model-based thinking activity also exploits *external models*. We have seen that the concept of manipulative abduction is devoted to capturing the role of action on external models in hypothetical and creative reasoning. This kind of manipulation provides otherwise unavailable information that enables the agent to solve a problem by performing abductive processes of generation or selection of hypotheses. An expert manipulation of objects directed by abductive movements that implicates the strategic application of old and new *templates* of behavior mainly connected with extra-theoretical components also esthetical, ethical, and emotional.

Manipulative abduction happens when we are thinking *through* doing and not only, in a pragmatic sense, about doing. It refers to an extra-theoretical be-

¹ Magnani (2001a, 2002) introduces the concept of theoretical abduction. He maintains that there are two kinds of theoretical abduction, “sentential”, related to logic and to verbal/symbolic inferences, and “model-based”, related to the exploitation of internalized models of diagrams, pictures, etc., cf. below in this paper.

² Manipulative abduction and epistemic mediators are introduced and illustrated in Magnani (2001a).

havior that aims at creating communicable accounts of new experiences to integrate them into previously existing systems of experimental and linguistic (theoretical) practices. Gooding (1990) refers to this kind of concrete manipulative reasoning when he illustrates the role in science of the so-called “construals” that embody tacit inferences in procedures that are often apparatus and machine based. The embodiment is of course an expert manipulation of objects in a highly constrained experimental environment, and is directed by abductive movements that imply the strategic application of old and new *templates* of behavior mainly connected with extra-theoretical components, for instance emotional, esthetical, ethical, and economic.

The hypothetical character of construals is clear: they can be developed to examine further chances, or discarded, they are provisional creative organization of experience and some of them become in their turn hypothetical “interpretations” of experience, that is more theory-oriented, their reference is gradually stabilized in terms of established observational practices. Step by step the new interpretation - that at the beginning is completely “practice-laden” - relates to more “theoretical” modes of understanding (narrative, visual, diagrammatic, symbolic, conceptual, simulative), closer to the constructive effects of theoretical abduction. When the reference is stabilized the effects of incommensurability with other stabilized observations can become evident. But it is just the construal of certain phenomena that can be shared by the sustainers of rival theories. Gooding (1990) shows how Davy and Faraday could see the same attractive and repulsive actions at work in the phenomena they respectively produced; their discourse and practice as to the role of their construals of phenomena clearly demonstrate they did not inhabit different, incommensurable worlds in some cases. Moreover, the experience is constructed, reconstructed, and distributed across a social network of negotiations among the different scientists by means of construals.

To illustrate this process - from manipulations, to narratives, to possible theoretical models (visual, diagrammatic, symbolic, mathematical) - in a previous work³ I have considered some observational techniques and representations made by Faraday, Davy, and Biot concerning Oersted’s experiment about electromagnetism. They were able to create consensus because of their conjectural representations that enabled them to resolve phenomena into stable perceptual experiences. Some of these narratives are very interesting.

2.2 Epistemic mediators

Recent research, taking an ecological approach to the analysis and design of human-machine systems, has shown how expert performers use action in everyday life to create an “external” model of task dynamics that

can be used in lieu of an internal model (Kirlik, 1998). Not only a way for moving the world to desirable states, action performs an *epistemic* and not merely performatory role that is very relevant to abductive reasoning.

The whole activity of manipulation is devoted to build various external *epistemic mediators* that function as an enormous new source of information and knowledge. I derive this expression from the cognitive anthropologist Hutchins (1995), that coins the expression “mediating structure” to refer to various external tools that can be built to cognitively help the activity of navigating in modern but also in “primitive” settings. Any written procedure is a simple example of a cognitive “mediating structure” with possible cognitive aims: “Language, cultural knowledge, mental models, arithmetic procedures, and rules of logic are all mediating structures too. So are traffic lights, supermarkets layouts, and the contexts we arrange for one another’s behavior. Mediating structures can be embodied in artifacts, in ideas, in systems of social interactions [...]” (290-291).

In this light manipulative abduction in science represents a kind of redistribution of the epistemic and cognitive effort to manage objects and information that cannot be immediately represented or found internally (for example exploiting the resources of visual imagery).⁴

2.3 Manipulative templates

The *hypothetical* character of manipulations in creativity is clear: they are a sort of test, they can be developed to examine further chances, they are a provisional creative organization of experience and some of them become in their turn hypothetical “interpretations” of experience, suggesting new worldviews. Step by step the new interpretation - that at the beginning is completely “practice-laden” - relates to more “theoretical” modes of understanding (narrative, visual, diagrammatic, symbolic, conceptual, simulative).

It is possible to have a better understanding by rendering explicit some of the manipulative *templates* of behavior that are active in creative abduction: 1. action elaborates a *simplification* of the reasoning task and a redistribution of effort across time when we “need to manipulate concrete things in order to understand structures which are otherwise too abstract” (Piaget 1974), or when we are in presence of *redundant* and unmanageable information; 2. action can be useful in presence of *incomplete* or *inconsistent* information - not only from the “perceptual” point of view; 3. action as a *control of sense data* illustrates how we can change the position of our body (and/or of the external objects) and how to exploit various kinds of prostheses

³ Cf. Magnani, 2002. (Cf. also Gooding, 1990).

⁴ For example it is difficult to preserve precise spatial relationships using mental imagery, especially when one set of them has to be moved relative to another.

(Galileo's telescope, technological instruments and interfaces) to get various new kinds of stimulation; 4. action enables us to build *external artifactual models* of task mechanisms instead of the corresponding internal ones, that are adequate to adapt the environment to agent's needs; 5. in science experimental action shows a sensibility to the aspects of the phenomenon which can be regarded as *curious* or *anomalous*; manipulations have to be able to introduce potential inconsistencies in the received knowledge; 6. action exhibits a preliminary sensibility to the *dynamical* character of the phenomenon, and not to entities and their properties, the common aim of manipulations is to practically reorder the dynamic sequence of events in a static spatial one that should promote a subsequent bird's-eye view (narrative or visual-diagrammatic).

3. A Cognitive Theory of the Abductive Modeling Activity

We can say abduction is a complex *process* that works through *imagination*: it suggests a new direction in reasoning by *shaping* new possible ways for explaining object and hypotheses (cf. the templates mentioned above). In this sense imagination should not be confused with an act of intuition. Peirce describes abduction as a dynamic modeling process that fluctuates between states of doubt and states of belief. To solve the doubt, and some eventually linked anomalies, the agent implements a process of information gathering which at the same time relates to the "problem", to the agent's evolving understanding of the situation and to its changing requirements. By imagination here I mean this process of knowledge gathering and shaping. A process, that Kant considered "blind", that leads to *see things as we would not otherwise have seen them*: "a blind but indispensable function of the soul, without which we should not have no knowledge whatsoever" (Kant, 1929, A78-B103, p.112). Scientific creativity, it is pretty obvious, involves seeing the world in a particular new way: scientific understanding permits us to see some aspects of reality in a particular way and creativity relates to this capacity to shed new light. Suggestions which make us able to further analyze this process come from a theory developed in the area of computer vision: the *active perception* approach (see Thomas, 1999).

This approach aims at understanding cognitive systems in terms of their environmental *situatedness*: instead of being used to build a comprehensive inner model of its surroundings, the agent's perceptual capacities are seen as simply used to obtain "whatever" specific pieces of information are necessary for its behavior in the world. The agent constantly "adjusts" its vantage point, updating and refining its procedures, in order to uncover a piece of information. This resorts to the need of specifying how to efficiently examine and explore and to the need of "interpreting" an object of a

certain type. It is a process of attentive and controlled perceptual exploration through which the agent is able to collect the necessary information: a purposefully moving through what is being examined, actively picking up information rather than passively transducing (cf. Gibson, 1979).

As suggested for instance by Lederman and Klatzky (1990), this view of perception may be applied to all sense modes: for example, it can be easily extended to the haptic mode. Mere passive touch, in fact, tells us little, but by actively exploring an object with our hands we can find out a great deal. Our hands incorporate not only sensory transducers, but musculature which, under central control, moves them in appropriate ways: lifting something tells about its weight, running fingers around the contours provides shape information, rubbing it reveals texture. As already stressed by Peirce in the quotation I already reported above, when dealing with the hypothesizing activity of what I call manipulative abduction, "A man can distinguish different textures of cloth by feeling: but not immediately, for he requires to move fingers over the cloth, which shows that he is obliged to compare sensations of one instant with those of another" (CP 5.221).

Thomas (1999) suggests we can think of the fingers together with the neural structures that control, for example, running them so that we can consider the afferent signals that they generate as a sort of (perceptual) *instrument* to gather knowledge: a complex of physiological structures capable of active testing for some environmental property. The study of manipulative abduction that I outlined above, can gain from this approach. To give an example, the role of particular epistemic mediators (*optical diagrams*) in non-standard analysis has been studied, and so their function in grasping and teaching abstract and difficult mathematical concepts (see Magnani and Dossena, 2002). In this case the external models (mathematical diagrams) do not give full available knowledge, but, on the contrary, compel the agent to engage a continuous epistemic dialogue between the diagrams and its internal knowledge to the aim of understanding an already existing information or at "creating" a new one (cf. also the geometrical example in the following section).

It is clear that humans and other animals make a great use of perceptual reasoning and kinesthetic abilities. We can catch a thrown ball, cross a busy street, read a musical score, go through a passage by imaging if we can contort our bodies to the way required, evaluate shape by touch, recognize that an obscurely seen face belongs to a friend of ours, etc. Usually the "computations" required to achieve these tasks are not accessible to a conscious description. Mathematical reasoning uses language explanations, but also non-linguistic notational devices and models. Geometrical constructions represent a relatively simple example of this kind of extra-linguistic machinery we know as

characterized in a model-based and manipulative - abductive - way.

3.1 An example: discovering new concepts with diagrams

Diagrams serve an important role in abduction because they can be manipulated. In mathematics diagrams play various roles in a typical abductive way. Two of them are central:

- they provide an intuitive *explanation* able to help the understanding of concepts difficult to grasp or that appear obscure and/or epistemologically unjustified,⁵
- they help to *create* new previous unknown concepts, as illustrated in the case of the non-Euclidean geometry.

In the case of the construction and examination of diagrams in geometrical reasoning, specific experiments serve as states and the implied operators are the manipulations and observations that transform one state into another. The geometrical outcome is dependent upon practices and specific sensory-motor activities performed on a non symbolic object, which acts as a dedicated external representational medium supporting the various operators at work. There is a kind of an epistemic negotiation between the sensory framework of the geometer and the external reality of the diagram. This process involves an *external representation* consisting of written symbols and figures that are manipulated "by hand". The cognitive system is not merely the mind-brain of the person performing the geometrical task, but the system consisting of the whole body (cognition is *embodied*) of the person plus the external physical representation. In geometrical discovery the whole activity of cognition is located in the system consisting of a human together with diagrams.

The external representations are not merely memory aids: they can give people access to knowledge and skills that are unavailable to internal representations, help researchers to easily identify aspects and to make further inferences, they constrain the range of possible cognitive outcomes in a way that some actions are allowed and other forbidden. The mind is limited because of the restricted range of information processing, the limited power of working memory and attention, the limited speed of some learning and reasoning operations; on the other hand the environment is intricate, because of the huge amount of data, real time requirement, uncertainty factors. Consequently, we have to consider the whole system, consisting of both internal

and external representations, and their role in optimizing the whole cognitive performance of the distribution

I stated above that in mathematics diagrams play various roles in a typical abductive way. Now I can add that:

- they are *epistemic mediators* able to perform various abductive tasks in so far as
- they are *external representations* which are devoted to provide explanatory abductive results.

Let us consider some aspects of the role of mirror and unveiling diagrams in the Lobachevskyan discovery of the elementary non-Euclidean geometry. A *mirror diagram* (for example the diagram of the drawn parallel lines - cf. Figure 1 - Lobachevsky, 1891) is a kind of external *analogous* both of the mental image we depict in the mental visual buffer and of the symbolic-propositional level of the postulate definition (the fifth postulate). In general this diagram mirrors the

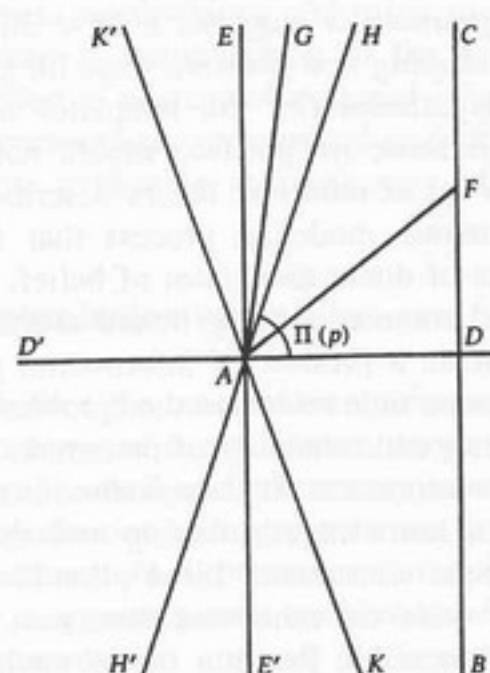


Figure 1.

internal imagery and provide the possibility of detecting anomalies. The external representation of geometrical structures often activates direct perceptual operations (for example, identify the parallels and search for the limits) to elicit consistency or inconsistency routines. Sometimes the mirror diagram biases are inconsistent with the task and so they can make the task more difficult by misguiding actions away from the goal. If consistent, they can make the task easier by guiding actions toward the goal. In certain cases the mirror diagrams biases are irrelevant, they should have no effects on the decision of abductive actions, and play lower cognitive roles. The in the case of Figure 1 the diagram of the parallel lines was used in the history of geometry to make both consistent and inconsistent the fifth Euclidean postulate and the new non-Euclidean perspective (more details on this epistemological situation – and on the role of what I call "mirror"

⁵ Some new optical diagrams (microscopes within microscopes), which provide new mental representations of the concept of tangent line at the infinitesimally small regions, are introduced in the already cited Magnani and Dossena (2002).

and “unveiling” diagrams in mathematical discovery are given in Magnani, 2001b and 2002b).

I have illustrated in the cited studies that some perceptions activated by the diagram are of course disregarded as irrelevant to the task, as it usually happens when exploiting external diagrammatic representations in reasoning processes. Because not everything in external representations is always relevant to a task, high level cognitive mechanisms need to use task knowledge (usually supplied by task instructions, geometrical, in our case) to direct attention and perceptual processes to the relevant features of external representations.

The important thing here is not just related to the fact that human beings simply have representations (they could have representations, for instance, about the history of painting, its products, colors, and techniques), but to the problem of “creatively” managing “representations”. It is important to examine how our *epistemic* instruments act in searching through the entire space of available knowledge to create new hypotheses and information. It seems that the agent does not first build up an inner representation of the surrounding environment (*its current situation*) and then it queries that: it queries the environment itself, when it needs specific information. The world is actively explored rather than passively registered.

It is this entire active process of epistemic negotiation that constitutes creativity and not the final product (a new model or theory) or the act of internal intuition. Creativity does not arise – so to say – when we actually see, but when we are *seeing nothing in particular* (something not yet defined and conceptualized) *as something*, whatever it may be. This certainly is an important aspect of scientific research: the ability to see things as we would not otherwise have seen them, and explore the world in new (creative) and useful ways (Tycho Brahe saw the Sun as rising over the horizon, whereas Kepler, a Copernican, saw the Earth turning towards the Sun).

4. Creativity and Ethical Mediators

The active process of information gathering through *mediators*, to shape knowledge, should not be restricted to the scientific activity and so to the “epistemic” side of them. An often neglected side of human creativity is, in fact, related to emotional, artistic, and ethical aspects, and concerns the active shaping of values in an esthetic or in an ethical world. In the case of morality the role of hypotheses and manipulation of the world is clear in Kant’s moral doctrine. When Kant considers pure moral rules, in fact, he says that they could be applied to the concrete experience through a kind of “typification”, a figurative envisioning of a non existing world, based on a metaphoric mapping, as a means for judging a given moral situation (Johnson, 1993).

As already outlined above, for Peirce all knowing is *inferring* and inferring is not instantaneous, it happens in a process that needs an activity of comparisons involving many kinds of models (signs) in a more or less considerable lapse of time. All sensations or perceptions participate in the nature of a unifying hypothesis, in the case of “emotions” too. In Peircian sense *emotions* too express a kind of model-based reasoning and have an “inferential” character. In decision making emotions play a distinguished role: they make the velocity of the decision process, surely related to what we care about, and lead directly to actions. But they are also usually considered irrational because of the serious disadvantages they present: failure to consider other options, lack of consideration of accurate and relevant information, not sharability in group situation, when the decisions have been adopted collectively. It is important to understand that emotions are not inherently irrational, for example they can be usefully intertwined with cultural aspects.

In general we can say that moral deliberations relate to a sort of selection or creation of principles (rules, prototypes) and to their application to concrete cases. We can both just select (or create, if we do not have any) moral principles (rules, prototypes) and apply them to concrete cases or looking for the best ones among them according to some ethical meta-criteria. When we create new ethics, we provide new knowledge and new rules about problems and situations not yet clearly covered from the moral point of view. In this last case we certainly are in front of a particular case, but the problem is not only the one of ethically solving the case at hand by applying already available ethical concerns – indeed we lack a satisfactory moral knowledge to handle the puzzling situation. Instead we need to create something new, for example new good reasons first which can provide an acceptable intelligibility of the problem. Once created, it will be possible to see the new principle and the new moral knowledge as a crystallization of the various insights emerging from peoples’ and/or experts’ experience and thinking.

The role of cognitive delegations to external objects and structures has to be extended to the case of human actions and organizations, so viewed as cognitive “mediating” mechanisms endowed with moral aspects. In this light it is possible to introduce the concept of *ethical* (or *moral*) *mediator*. Moral mediators play an important role in reshaping ethical worth of human beings and collectives. They especially involve a continuous reconfiguration of social orders aimed at rebuilding new moral perspectives and chances. These mediators represent a kind of redistribution of the moral effort through managing objects and information in such a way that we can overcome the poverty and the unsatisfactory character of the moral options immediately represented or found internally. *Moral mediators* are also used to exploit latent constraints in the human-environment system. These new constraints grant additional and precious ethical information. When we

spontaneously act in a way so that we spend more quality time with our partner to save our marriage, for example, then our actions automatically can cause variables relating to “unexpected” and “positive” contents of the relationship to covary with perceptible new released informative, sentimental, sexual, and in general bodily variables. Prior to the adoption of the new reconfigured “social” order of the couple, there is no active constraint between these hidden and overt variables causing them to carry information about each other. It is also well-known that also “trained” emotions⁶ play an important creative role in moral deliberations.

4.1 Templates of moral doing

It is difficult to establish a list of invariant behaviors that are able to illustrate manipulative reasoning in ethics. As illustrated above, certainly the expert manipulation of non-human objects in real or artificial environments implies the application of old and new *templates* of behavior that exhibit some regularities. As I have said it is important to remember they are embodied and implicit, as tacit forms of acting: I am not referring here to the moral actions and manipulations that simply follow previous explicit and devised plans. Anyway, this moral activity is still conjectural: these templates are embedded hypotheses of moral behavior (creative or already cognitively present in the people’s mind-body system, and ordinarily applied) that enable a kind of moral “doing”. Hence, some templates of action and manipulation can be *selected* in the set of those available and pre-stored, others have to be *created* for the first time to perform the most interesting accomplishments of manipulative moral inference.

Some common features of these “tacit” templates that enable us to manipulate external human and non-human things and structures to achieve moral effects are related to (Figure 2): 1. sensibility to the aspects of the moral situation which can be regarded as *curious* or *anomalous*; manipulations can also be performed to be able to introduce potential inconsistencies in the received knowledge (we suddenly adopt a different attitude with respect to our wife/husband to get some reactions we can regard as interesting – or “unexpected” – to confirm or discard hypotheses about her/feelings or to develop further hypotheses about them; in an investigation about a crime we spontaneously engage further manipulations of the evidence to get more interesting data to morally shape the suspect); 2. preliminary sensibility to the *dynamical* character of the situation at hands, and not only to entities and their properties, common aim of manipulations is to practically reorder the dynamic sequence of the events correlated to the main problem to promote the subsequent possi-

bility of new possibilities and options for action (a women in front of decision in favor of abortion spontaneously tries to modify the dynamical aspects of her behavior and the structure of her human relationships to try to establish new perspectives able to make her able to envisage a possible decision different from the first one first envisaged); 3. referral to manipulations that exploit *artificial* created feelings and environments to free new possibly stable and repeatable sources of information about hidden moral knowledge and constraints (when dealing with the moral problem of capital punishment we can spontaneously handle people, for example with statistics, interviews, scientific research, associations, to artificially reconfigure social orders⁷ in a way suitable to get real and not hypocritical information, for example about the real relief generated in the victim’s relatives by killing the criminal); 4. various contingent ways of spontaneous moral acting: *looking* from different perspectives, *checking* the different information available, *comparing* subsequent events, *choosing*, *discarding*, *imaging* further manipulations, *re-ordering* and *changing relationships* in the world by implicitly *evaluating* the usefulness of a new order (for instance, to help memory) (in the ethical case they certainly are all useful ways for getting suitable evidence and for stimulating the derivation of further consequences to test our previously established moral judgments; analogous of all these manipulative templates are active in epistemic settings, as illustrated in Magnani, 2001a and Magnani, Piazza, and Dossena, 2002).

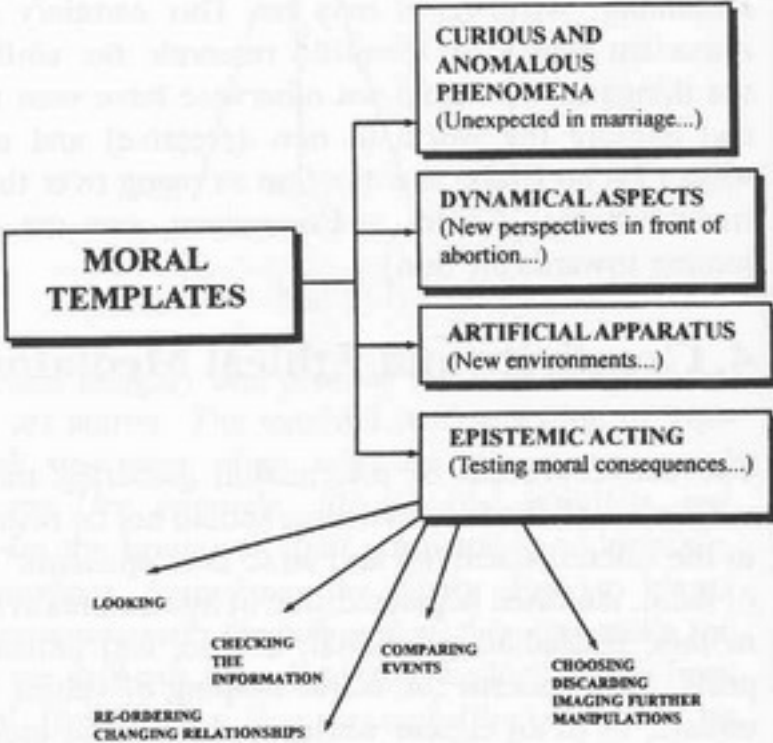


Figure 2.

⁶ That is not just shaped by biological evolution but also by cultural aspects.

⁷ On the reconfiguration of social orders that is realized in science laboratories cf. Knorr Cetina, 1999.

More features of our tacit templates and ethical mediators are related to the following additional issues (Figure 3): 5) moral spontaneous action that can be useful in presence of *incomplete* or *inconsistent* information – not only from the “perceptual” point of view – or of a diminished capacity to morally act upon the world: it is used to get more data to restore coherence and/or to improve deficient knowledge; 6) action as a *control of sense data* illustrates how we can change the position of our body (and/or of the external objects) to reconfigure social orders, collective relationships, and how to exploit various kinds of artificially created events to get various new kinds of stimulation: action provides some tactile, visual, kinesthetic, sentimental, emotional, and bodily information (e.g. in taking care of people, cf. below in the following subsection), otherwise unavailable; 7) action enables us to build new *external artifactual models* of ethical mechanisms and structures (for example through “institutions”) instead of the corresponding “real” and “natural” ones.⁸ For instance, we can substitute to the “natural” structure “family” an environment more adequate to agent’s moral needs. In this case we aim at reconfiguring relationships for instance when we exploit the social reshaping role of the “houses” were children molested inside family are recovered, to rebuild in a whole artificial framework their moral perception for example of the sexual molestation received and of the related bad feelings. Something similar occurs in the case of the addicted people. We also establish structures to implicitly favor good manners, for example fences, barriers in the lines, etc.

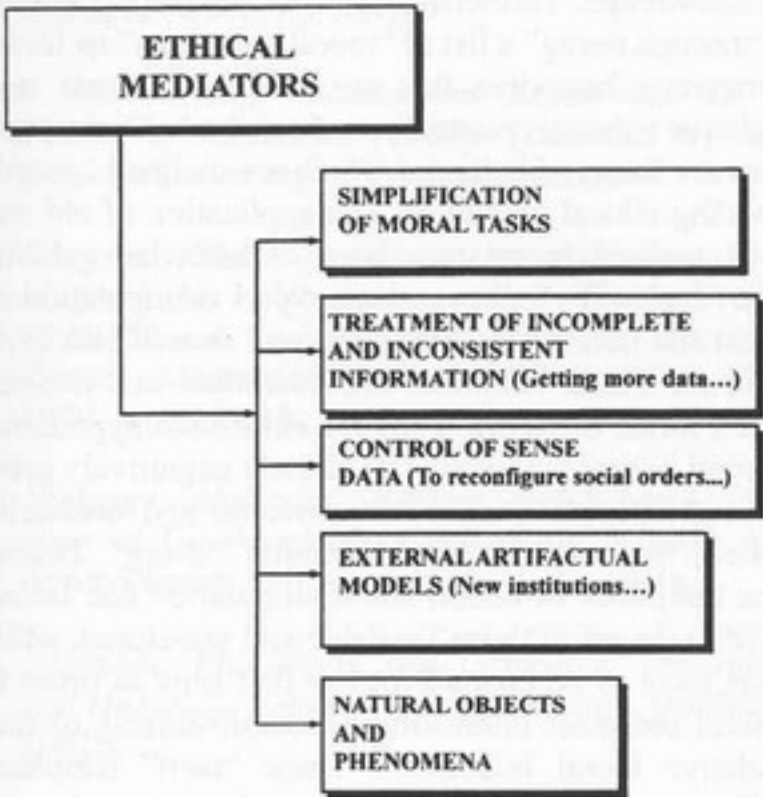


Figure 3.

⁸ Of course these “real” and “natural” structures are also artificial, because we can think of a “family” as a kind of not merely natural institution.

4.2 Moral mediators

The whole activity of manipulation is also devoted to build various external *moral mediators*⁹ that function as an enormous new source of information and knowledge. Therefore, these mediators represent a kind of redistribution of the moral effort through managing objects and information in such a way that we can overcome the poverty and the unsatisfactory character of the moral options immediately represented or found internally (for example exploiting the resources in terms of merely internal/mental moral principles, utilitarian envisaging, and model-based moral reasoning). Not only a way for moving the world to desirable states, action performs a moral and not just merely performatory role: people structure their worlds to simplify and solve moral tasks when they are in presence of incomplete information or possess a diminished capacity to morally act upon the world when they have insufficient opportunities to know. *Moral mediators* are also used to exploit latent constraints in the human-environment system. These elicited new constraints grant us additional and precious ethical information: when we spontaneously act in a way in which we spend more quality time with our partner to save our marriage, then our actions automatically cause variables relating to “unexpected” and “positive” contents of the relationship to covary with perceptible new released informative, sentimental, sexual, and, in general, bodily variables. Prior to the adoption of the new reconfigured “social” order of the couple, there is no active constraint between these hidden and overt variables causing them of carry information about each other.

Also natural phenomena can play the role of external artifactual moral mediators: I am referring to the role played by external “natural” objects as occasions for suggesting new creative ethical knowledge, like in the case of endangered species and “cognitive” tools. Many external things that usually are (or in the past were) inert from the moral point of view can be transformed into moral mediators. For example we can use animals to depict new moral features of living objects previously unseen, so we can do with earth or (non natural) cultural objects; we can also use external “tools” like writing, narratives, others persons’ information, rituals, various kinds of pertinent institutions to reconfigure previously given social orders morally unsatisfactory. Hence, not all of the moral tools are inside the head, many of them are shared and distributed in external objects and structures which function as ethical devices.

The external moral mediators are endowed with functional properties as components of a memory sys-

⁹ I derive this expression from the one “epistemic mediators” I introduced in Magnani (2001a, chapter 3): these consist of external representations, objects, and artifacts that are relevant in scientific discovery and reasoning processes.

tem crossing the boundary between person and environment (for example they are able to transform the tasks involved in allowing simple manipulations that promote further moral inferences at the level of model-based abduction): I can only enhance my bodily experience of pain through action by following the template *control of sense data*, I previously outlined, that is through changing – unconsciously – the position of my body and its relationships with other humans and non-humans embedded in distressing experiences. In many people moral training is often related to these kinds of spontaneous (and “lucky”) manipulations of the control of their own body and sense data so that they build their morality immediately and non reflectively “through doing”. Women taught all human beings the importance of attitudes that emphasize intimacy, caring and personal relationships.¹⁰ It would seem that women’s basic moral orientation is “taking care” of others and of external things in a personal way, not just being concerned by the humanity or by the world in general. The ethics of care does not consider “obligation” as essential; moreover, it does not require that we impartially promote the interests of everyone alike. It looks on small-scale relationships with people and objects, so that it is not important to “think” to help disadvantaged children in all the world (like men aim at doing) but to “do” that when needed, just “over there”.

In light of my philosophical and cognitive treatment of the problem of moral model-based thinking and of morality “through doing”, this female attitude does not have to be considered as less rational and deontological because more related to emotions and passions, in turn intended as forms of lower level of thinking. I contend that we can become “more loving parents” in a more intuition and feeling oriented way because of the reasons – and may be because of “Kantian” rules – that compel us to do educate our feelings in that way, and so to privilege the “taking care” of our children in certain situations. The route from reason to feeling (and of course also from feeling to reason) is continuous in ethics.

Consequently, “taking care” is a “different” way of looking at people and objects, and, as a form of morality immediately given “through doing”, is clearly integrated and explained as a fundamental kind of moral inference and knowledge. Ethics of care relates to the habit of terminating some people’s urgencies by performing a “caring perception” of non-humans (objects inside the house, for example). Consequently, these non-humans can easily be seen as “moral mediators” in the sense that I give to this cognitive concept.

¹⁰ On the role of feminist skepticism in ethics, and the so-called “expressive-collaborative model” of morality cf. Urban Walker, 1996. This model looks at moral life as “a continuing negotiation among people, a socially situated practice of mutually allotting, assuming, or deflecting responsibilities of important kinds, and understanding the implications of doing so” (p. 276). Or course this is contrasted to the so-called “theoretical-juridical conception of morality”.

3. Conclusion

What I call *theoretical abduction* (sentential and manipulative) certainly illustrates much of what is important in creative abductive reasoning both in humans and computational programs, especially the objective of selecting and creating a set of hypotheses that are able to dispense good (preferred) explanations of data, but fails to account for many cases of explanations occurring in science or in everyday reasoning when the exploitation of the environment is crucial. The concept of *manipulative abduction* is devoted to capture the role of action in many interesting situations: action provides otherwise unavailable information that enables the agent to solve problems by starting and performing a suitable abductive process of generation or selection of hypotheses. Many external things, even if usually inert from the epistemological point of view, can be transformed into what is called *epistemic mediators*, which are illustrated in the second part of this paper, together with an analysis of the related notion of “external representation steps in a way that discharges the “internal” mind of a computational load. To define a cognitive system it seems we can no longer identify it only with internal processing devices.

By exploiting the concept of “thinking through doing” and of manipulative abduction I have tried to shed new light on some of the most interesting cognitive aspects of creative ethical reasoning of what I call “ethical mediators”. Indeed, I contend that the whole activity of manipulation can be seen as an activity for building various external “ethical mediators” that function as an enormous new source of information and knowledge. Furthermore, while describing morality “through doing” a list of “moral templates” as forms of invariant behaviors that are able to illustrate manipulative ethical reasoning is furnished. These templates are forms of behavior which are inclined towards providing ethical outcomes. The application of old and new (creative) moral templates of behavior exhibits some regularities and expresses expert manipulation of human and non-human objects in real or artificial environments. These templates are embodied and implicit as tacit forms of acting. They are embedded hypotheses of moral behavior (creative or already cognitively present in the people’s mind-body system, and ordinarily applied) that enable a kind of moral “doing”. Hence, some templates of action and manipulation can be selected in the set of those available and pre-stored, while others have to be created for the first time in order to perform the most interesting accomplishments of manipulative moral inferences. These “tacit” templates enable us to manipulate external human and non-human things and structures to achieve moral effects.

References

- M. A. Boden. *The Creative Mind: Myths and Mechanisms*. Basic Books, New York, NY, 1991.

- B. G. Buchanan. Creativity at the metalevel. AAAI-2000 presidential address. *AI Magazine*, fall 2001:13-28.
- J.J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, MA, 1979.
- D. Gooding. *Experiment and the Making of Meaning*. Kluwer, Dordrecht, 1990.
- E. Hutchins. *Cognition in the Wild*. MIT Press, Cambridge, MA, 1995.
- M. Johnson. *Moral Imagination. Implications of Cognitive Science in Ethics*. The University of Chicago Press, Chicago, 1993.
- I. Kant. *Critique of Pure Reason*, translated by N. Kemp Smith, MacMillan, London, 1929. Reprint 1998. Originally published 1787.
- K. Knorr Cetina, *Epistemic Cultures. How the Sciences Make Knowledge*. Harvard University Press, Cambridge, MA, 1999.
- A. Kirlik. The ecological expert: acting to create information to guide action. In *Proceedings of the 1998 Conference on Human Interaction with Complex Systems (HICS'98)*, IEEE Press, Piscataway, NJ, 1998.
- S. J. Lederman and R. Klatzky. Haptic exploration and object representation. In M.A. Goodale (ed.), *Vision and Action: The Control of Grasping*, Ablex, Norwood, NJ, 1990, pp. 98-109.
- N.J. Lobachevsky. *Geometrical Researches on the Theory of Paralleles* [1840]. Translated by G.B. Halsted. University of Texas, Austin, 1891.
- L. Magnani. Model-based creative abduction. In L. Magnani, N. J. Nersessian, and P. Thagard (eds.), *Model-Based Reasoning in Scientific Discovery*. Kluwer Academic/Plenum Publishers, New York, 1999, pp. 219-238.
- L. Magnani. *Abduction, Reason, and Science. Processes of Discovery and Explanation*. Kluwer Academic/Plenum Publishers, New York, 2001a.
- L. Magnani. *Philosophy and Geometry. Theoretical and Historical Issues*. Kluwer Academic, Dordrecht, 2001b.
- L. Magnani. Thinking through doing, external representations in abductive reasoning, in *AISB 2002 Symposium on AI and Creativity in Arts and Science*, London, Imperial College, 2002a.
- L. Magnani. Epistemic mediators and model-based discovery in science. In: L. Magnani and N.J. Nersessian (eds.), *Model-Based Reasoning: Science, Technology, Values*. Kluwer Academic/Plenum Publishers, New York, 2002b, pp. 305-329.
- L. Magnani and R. Dossena. Perceiving the infinite and the infinitesimal world: unveiling and optical diagrams and the construction of mathematical concepts. To appear in *Foundations of Science*, 2002.
- L. Magnani and N.J. Nersessian (eds.). *Model-Based Reasoning: Science, Technology, Values*. Kluwer Academic/Plenum Publishers, New York, 2002.
- L. Magnani, M. Piazza, and R. Dossena. Epistemic mediators and chance morphodynamics. In: A. Abe (ed.), *Proceedings of PRICAI-02 Conference, Working Notes of the 2nd International Workshop on Chance Discovery*, Tokyo, pp. 38-46.
- N. J. Nersessian. Should physicists preach what they practice? Constructive modeling in doing and learning physics. *Science and Education*, 4: 87-120, 1995.
- C. S. Peirce. *Collected Papers 1-6 (CP)*. Edited by C. Hartshorne and P. Weiss. *Collected Papers 7-8*. Edited by A. Burks. Harvard University Press, Cambridge, 1931-35, 1958.
- J. Piaget. *Adaptation and Intelligence*. University of Chicago Press, Chicago, IL, 1974.
- Sternberg, R. J. (ed.). *Handbook of Creativity*. Cambridge University Press, Cambridge, 1999.
- R. J. Sternberg, J. C. Kaufman, and J. E. Pretz (eds.). *The Creativity Conundrum: A Propulsion Model of Kinds of Creative Contributions*. Psychology Press, New York, 2002.
- N. J. Thomas. Are theories of imagery theories of imagination? An active perception approach to conscious mental content, in *Cognitive Science*, 23 (2): 207-245, 1999.
- M. Urban Walker, Feminist skepticism, authority and transparency. In W. Sinnott-Armstrong and M. Timmons (eds.), *Moral Knowledge? New Readings in Moral Epistemology*, Oxford University Press, Oxford 1996, pp. 267-292.

Music Generation from Statistical Models

Darrell Conklin

Department of Computing
City University
Northampton Square
London EC1V 0HB

Abstract

This paper discusses the use of statistical models for the problem of musical style imitation. Statistical models are created from extant pieces in a stylistic corpus, and have an objective goal which is to accurately classify new pieces. The process of music generation is equated with the problem of sampling from a statistical model. In principle there is no need to make the classical distinction between analytic and synthetic models of music. This paper presents several methods for sampling from an analytic statistical model, and proposes a new approach that maintains the *intra opus* pattern repetition within an extant piece. A major component of creativity is the adaptation of extant art works, and this is also an efficient way to sample pieces from complex statistical models.

1 Introduction

In his *Syntactic Structures* Chomsky (1957) used the term *creativity* to refer to the unique capacity of humans to understand and produce an indefinitely large number of sentences in a language, most of which have never been encountered or spoken before. The goal of a linguistic theory was the formulation of general principles for the evaluation of alternative grammars that could account for human creativity.

The construction of computational methods for musical style imitation has been far more difficult than initially imagined. Pioneering activity and early hope in the 1950s (Pinkerton, 1956; Brooks et al., 1956; Hiller and Isaacson, 1959), driven by advances in machine learning and information theoretic musicology (Meyer, 1956; Cohen, 1962), was soon replaced by frustration as the models used proved incapable of generating even simple well-formed melodies. This naturally led to the shift of research to hand-crafted algorithms for style imitation that had only small explicit statistical and empirical components (Lidov and Gabura, 1973; Baroni and Jacobini, 1978; Cope, 1991; Sundberg and Lindblom, 1991). While unquestionably successful, these models are heavily biased by their developers, and to the extent that they constrain the set of possible musical productions, are not robust and do not exhibit creativity. The promise of robust, creative, style-independent empirical learning methods for musical style imitation has not yet been fulfilled.

The study of grammars for music has a long history that has in many ways paralleled that of natural language. Chomsky's (1957) criticism of Markov (or finite-state, or n-gram) models for language proved to be damaging to both areas. Chomsky's main position was that a) finite-state methods could not capture, in a compact fashion,

non-adjacent dependencies between words in sentences; b) they could not model the recursive embedding of phrase structure found in natural language; and therefore c) grammatical sentence production cannot be achieved by a left-to-right process. Though the relative impact of these points can be debated with respect to music, the effect in the 1970s and 1980s was to initiate the development of more powerful grammars for music generation, while at the same time suppressing the high empirical component of earlier work. In the 1990s work in computational language modeling, motivated by the applied task of speech recognition, made a dramatic shift back to statistical models (Jelinek, 1997). Improved methods for language modeling were developed that extended the basic n-gram model while maintaining its tractability and simplicity. Research in computational music modeling, partly driven by the growth of on-line music databases, also made this shift (Conklin and Witten, 1995; Ponsford et al., 1999), and also expanded into polyphonic music.

In other ways, computational modeling of music and natural language have taken different directions in their intended application. Natural language modeling was focused from the beginning of the rationalist program of the 1960s on developing *analytic* models — those intended to classify sentences as grammatical or non-grammatical. The main goal was thereby to demonstrate linguistic *competence*, while operational issues of sentence generation and production were relegated to the topics of *performance* and *pragmatics*. Motivated by astounding successes at speech recognition, computer scientists have recently been called upon to develop the best analytic models possible for natural language; the topic of sentence production has been confined to a small dedicated group of natural language generation researchers.

By contrast, in music, the situation has been nearly

reversed. The focus from the early years was on *synthetic* models — those intended to generate well-formed sentences from a grammar. This is probably because the practical application of analytic models has not, until recently, been so clear as it has for statistical language models. The generation of music, yielding immediate results that can be listened to and evaluated by the researcher, is an exciting and compelling research project. The wholly subjective evaluation of results, however, is fraught with methodological problems (Pearce and Wiggins, 2001). Only recently have researchers in music informatics turned their attention to the analytic topics of music prediction (Conklin and Witten, 1995; Reis, 1999; Triviño-Rodríguez and Morales-Bueno, 2001; Pearce, 2003), phrase structure analysis (Bod, 2002), and music classification (Westhead and Smaill, 1993; Sawada and Satoh, 2000; Cruz-Alcázar and Vidal-Ruiz, 1997; Dubnov et al., 1998; Chai and Vercoe, 2001). A common thread to these analytic studies is the importance given to objective measures with which alternative models of music can be compared and ranked.

The thesis of this paper is that the topics of creative music generation and analysis are in fact highly interconnected, and that in principle there is no need to make the classical distinction (Ruwet, 1966) between analytic and synthetic models of music. Analytic statistical models have an objective goal which is to assign high probability to new pieces in a style. These models can guide the generation process by evaluating candidate generations and ruling out those with low probabilities. The generation of music is thereby equated with the problem of sampling from a statistical model, or equivalently, exploring a search space with the statistical model used for evaluation. This paper presents several ways that this sampling may be performed. Perhaps surprisingly, only a few of the proposed sampling methods have been explored in the music generation literature. For example, sampling may be performed in a way that generates incremental modifications to extant pieces in a style, retaining their overall repetition structure during the process. In this way, an obvious feature of human creativity — the use and adaptation of existing pieces — may be computationally explored.

This paper is structured as follows. In Section 2 a review of statistical models of music is provided. Particular emphasis is given to the prevalent class of *history-based* models, which condition events in music based on features computable from preceding events. In Section 3 methods for music generation from statistical models are outlined. The music generation process is equated with the problem of sampling high probability pieces from a statistical model. The prevalent random walk methods are discussed and shown to be flawed in that they do not guarantee that pieces with high overall probability will be produced. For certain simple types of statistical model, it is possible to compute the optimal pieces from the model. For complex models, computing the optimal pieces can

be infeasible, and heuristic search or stochastic sampling is required. Modifications to extant pieces in a style is a very efficient way to sample high probability pieces from a model. The section concludes with an outline of a sampling approach that maintains the *intra opus* pattern repetition within an extant piece in a style.

2 Statistical models of music

A piece of music is represented by a sequence of *events*, which are music objects together with a duration and an onset time after instantiation into a piece. Notes are music objects, as are (recursively) sequences or simultaneities of music objects (Conklin, 2002). In addition to linear melody, this scheme allows for the representation of phrase structure and polyphony.

A statistical model of music assigns to every possible piece of music a probability. A statistical model captures regularities in a *class* of music, be this a genre, a style, a composer's style, or otherwise. A good model assigns high probabilities to pieces in the class, and low probabilities to pieces that are not in class. The probability of a piece p according to a model m is denoted $P(p | m)$.

In the Bayesian classification framework, there are several statistical models m_i , each of a different class i . A piece p is classified into the class i for which $P(p | m_i)$ is the highest (this assumes equal prior probabilities on all models, a detail which is not of concern in this paper). One of these models may be a *null model* which assigns equal probabilities to all valid pieces and is not permitted to assign a piece a higher probability than its true class model.

Statistical models of music are created empirically by induction. A corpus of training pieces in a class is used to instantiate the parameters of a statistical model. Competing models for the same class are evaluated according to their performance on a blind test set, or alternatively, by simulation using a cross-validation technique.

2.1 Context models

The most prevalent type of statistical model encountered for music, both for analysis and synthesis, are models which assign probabilities to events conditioned only on earlier events in the sequence. By the general term *context model*, we intend to include Markov, hidden Markov, n-gram, and finite state models. There are several reasons for the prevalence of context models in the music generation literature:

- events in a piece can be sometimes be predicted from preceding events;
- context models are easy to induce from examples: this typically involves storing subsequences in a data structure providing rapid access (such as a suffix tree);

- context models are usually very fast: efficient data structures can be used to rapidly match contexts. This allows their use in real-time algorithmic composition systems;
- the probability of a piece according to a context model is easily computed, being simply the product of probabilities of events in the sequence;
- it is straightforward to generate new music with a context model.

Models which employ highly specific contexts to make predictions are bound to fail when applied to new music, because training corpora are always limited in size and few sequences of events will be encountered *inter opus*. Solutions to this *sparse data* problem range from simple data preprocessing steps such as transposition to a common key, to *smoothing* of short and long contexts (Jelinek, 1997) and the related PPM technique (Conklin and Witten, 1995).

2.2 Complex statistical models

The sparse data problem recalls the nativist position that children cannot possibly learn grammars from the limited amount of primary data encountered. Chomsky (1957) argued for innate human capacity and cognitive structures, with learning being more a form of abductive grammar selection rather than inductive inference from examples. In a piece of music, events can be ascribed properties, encoded into and derived from background knowledge, and these properties will be far less numerous than actual events. The method of *viewpoints* (Conklin and Witten, 1995; Conklin and Anagnostopoulou, 2001; Conklin, 2002) therefore deals with the sparse data problem by applying knowledge of music. Events are predicted by an interpolation of the predictions of multiple viewpoints. Similar ideas have been employed in statistical natural language modeling (Brown et al., 1992), where context models over lexical categories are developed.

The technique of viewpoints is an instance of the general class of *history-based* models, wherein any features computable from preceding events can be used to condition the probability of the current event. Furthermore, the current event can be predicted by an interpolation of two separate models: one which captures short-term phenomena specific to the current piece, and a long-term model which captures regularities of the general style (Conklin and Witten, 1995).

Statistical models are not restricted to general history-based models. More powerful grammars in the Chomsky hierarchy, such as context-free grammars, can have a statistical interpretation. Learning the parameters of these statistical models, however, presupposes an existing grammar for a musical style. A type of context-free grammar, called the *dependency grammar* (Chelba et al., 1997), may have promise for music. This type of model

could elegantly capture dependencies between, for example, chord and non-chord tones in tonal music.

In a final analysis, however, music requires features even beyond the capabilities of context-free grammars. For example, the simple phenomenon of a repeated phrase in a piece cannot adequately be expressed using a context-free grammar. The sampling techniques discussed below for the generation of music from statistical models are applicable regardless of the power and complexity of the statistical model. The next section will outline a way to preserve pattern repetition in a piece.

3 Generation of music from statistical models

As outlined in the previous section, the goal of an analytic statistical model of a corpus of music is to assign high probabilities to pieces in the class, and lower probabilities to all other pieces. In the Bayesian framework, given multiple class models, a piece is classified by the model which assigns it the highest probability.

To generate a piece from an analytic model is therefore to sample a piece which has a high probability according to the model; presumably higher than its probability according to competing models. It is important to note that a high probability piece need not comprise only high probability events.

3.1 Random walk method

The simplest way to generate music from a history-based model is to sample, at each stage, a random event from the distribution of events at that stage. After an event is sampled, it is added to the piece, and the process continues until a specified piece duration is exceeded. This method has been employed by many, if not most, synthetic models of music (Brooks et al., 1956; Mozer, 1994; Conklin and Witten, 1995; Ponsford et al., 1999).

The random walk method, while applicable for real-time music improvisation systems that require fast and immediate system response (Assayag et al., 1999; Pachet, 2002), is flawed for generating complete pieces because it is “greedy” and cannot guarantee that pieces with high overall probability will be produced. The method may generate high probability events but may at some stage find that subsequently only low probability events are possible, or equivalently, that the distribution at subsequent stages have high entropy.

Allan (2002) provides a demonstration of this effect in music. The probability of the optimal sequence of harmonic symbols generated from a hidden Markov model (see next section) is compared to a sequence produced by the random walk method. The probability of the sequence produced can be significantly lower with the random walk method.

3.2 Hidden Markov models and Viterbi decoding

A *hidden Markov model* (HMM) is a statistical model in which *observed events* are generated from underlying *hidden states*. After generating an event from a state, a model moves into a new state based on its transition probabilities. The term *hidden* derives from the fact that many different state sequences can generate the same observed sequence of events. Different state sequences can therefore have different probabilities.

A key concept of HMM theory is the *decoding* step: given a sequence of observed events, find the most probable hidden state sequence. For a first-order HMM, this decoding can be computed by a dynamic programming algorithm (called the Viterbi algorithm) in time proportional to the length of the observed event sequence times the number of states in the model. A similar dynamic programming algorithm can be applied to non-hidden first-order Markov models to find the most probable sequence.

HMMs have recently been applied with success to music generation. Farbood and Schoner (2001) describe a system for producing a counterpoint line to a cantus firmus in the style of Palestrina. For a given cantus firmus, the most probable counterpoint line is found by Viterbi decoding. Allan (2002) describes an HMM approach for chorale melody harmonization, where Viterbi decoding is used to produce the most probable underlying sequence of harmonic symbols for a given melody line.

3.3 Stochastic sampling

A drawback of Viterbi decoding is that its computation time increases exponentially with the context length of the underlying Markov model on states. Furthermore, more complex and powerful statistical models are not readily transformed into a form in which Viterbi decoding is applicable. Producing the highest probability pieces from complex statistical models is therefore a computationally expensive task, and heuristic search and control strategies must be applied. With an increase in computation time, techniques such as A^* search can be used to compute the n best state sequences (Jiménez et al., 1995). However, a system that produces only a few high-probability pieces obviously cannot be called creative, and more robust methods are required.

Sampling from complex statistical models can be performed in various ways (Rosenfeld et al., 2001). To generate a piece using *Gibbs sampling* from a model m , we start with some initial piece p . The following process is iterated: a random event of the piece is chosen and all valid events are substituted into that position, each producing a new piece p' . One such piece p' , having the probability $P(p' | m)$, is chosen at random for the next iteration.

The Gibbs sampling method can be slow because the term $P(p' | m)$, which might itself be expensive to compute, must be computed many times. In *Metropolis sam-*

pling from a model m , a random event in the piece p is chosen, and a *single* event is substituted into that position, producing a new piece p' . This piece is accepted for the next iteration if $P(p' | m) > P(p | m)$, and otherwise rejected with a rejection probability that increases with each iteration.

Both sampling methods discussed above can fall into local valleys, where valid substitutions at all locations cannot improve the probability of the current piece. Both methods can be assisted by providing a high probability extant piece as a starting point. This brings up the interesting fact that modifications to extant pieces in a style is a very efficient way to generate high probability pieces in a style.

3.4 Pattern-based sampling

The application of statistical sampling techniques to music faces two main challenges. First, it is unlikely that single-event substitutions can provide the necessary diversity to exhibit creativity. Motif or phrase-level substitutions seem to be required, and this again raises the issue of the high computation time needed to compute all valid motifs for substitution. This limitation is beyond the scope of this paper. Second, musical cohesion (Anagnostopoulou, 1997) is not preserved by the sampling procedures. For example, in a piece in which repetition of a certain pattern should be preserved a substitution of an event within one occurrence of the pattern should also be reflected in the other.

An early attempt to handle *intra opus* repetition with a complex statistical model involved the use of a short-term model that adapted to the current piece being generated (Conklin and Witten, 1995). A flaw with this approach is that while pattern continuation can easily be handled, there is no way to specify at the outset of generation where repeated patterns should begin and end. One way to handle this effect is to apply modern pattern discovery algorithms (Cambouropoulos, 1998; Roland and Ganascia, 2000; Conklin and Anagnostopoulou, 2001) which can discover *intra opus* repetition, often at deeper levels than the basic musical surface.

A pattern-based sampling approach can first apply a pattern discovery algorithm to an extant piece to reveal patterns at various levels of abstraction. During stochastic sampling, the discovered pattern structure can be conserved. This is one way to overcome the limitations with n -gram models of music discussed earlier.

4 Conclusions

Analytic statistical models have an objective goal which is to construct models for a stylistic corpus that assign high probability to new pieces in the style. This essay has argued for increased attention to analytic models of music. Operational issues of music generation, expressed as the problem of sampling from a statistical model, can

be separated from the issues of model selection and training. The generation of music should not be confined to the prevalent random walk method for sampling from a history-based model. With music generation rephrased as a classical search and sampling problem, generation algorithms are free to apply deep knowledge and draw from extant pieces to reduce the search space and more rapidly focus on high probability pieces. The generation of music can use modern pattern discovery methods applied to extant pieces to reveal their repetition structure and provide musical cohesion to new productions. A major component of creativity is the adaptation of extant art works, and this is also an efficient way to generate music from complex statistical models.

References

- M. Allan. Harmonising Chorales in the Style of J.S. Bach. Master's thesis, School of Informatics, University of Edinburgh, 2002.
- C. Anagnostopoulou. Lexical cohesion in linguistic and musical discourse. In *Proc. 3rd ESCOM*, 1997.
- G. Assayag, S. Dubnov, and O. Delerue. Guessing the composer's mind: applying universal prediction to musical style. In *Proceedings of the International Computer Music Conference*, Beijing, China, 1999. International Computer Music Association.
- M. Baroni and C. Jacobini. *Proposal for a Grammar of Melody*. Les Presses de l'Université de Montréal, 1978.
- R. Bod. Memory-based models of melodic analysis: Challenging the gestalt principles. *Journal of New Music Research*, 31(1):27–37, 2002.
- F. P. Brooks, A. L. Hopkins Jr., P. G. Neumann, and W. V. Wright. An experiment in musical composition. *IRE Transactions on Electronic Computers*, EC-5:175–182, 1956.
- P. Brown, V. Della Pietra, P. deSouza, J. Lai, and R. Mercer. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479, 1992.
- E. Cambouropoulos. *Towards a General Computational Theory of Musical Structure*. PhD thesis, University of Edinburgh, 1998.
- W. Chai and B. Vercoe. Folk music classification using hidden Markov models. In *Proc International Conference on Artificial Intelligence*, 2001.
- C. Chelba et al. Dependency language modeling. Technical Report Research Note 24, Center for Language and Speech Processing, The Johns Hopkins University, 1997.
- N. Chomsky. *Syntactic Structures*. Mouton, 1957.
- J. E. Cohen. Information theory and music. *Behavioral Science*, 7:137, 1962.
- D. Conklin. Representation and discovery of vertical patterns in music. In C. Anagnostopoulou, M. Ferrand, and A. Smaill, editors, *Music and Artificial Intelligence: Lecture Notes in Artificial Intelligence 2445*, pages 32–42. Springer-Verlag, 2002.
- D. Conklin and C. Anagnostopoulou. Representation and discovery of multiple viewpoint patterns. In *Proceedings of the International Computer Music Conference*, pages 479–485, Havana, Cuba, 2001. International Computer Music Association.
- D. Conklin and I. Witten. Multiple viewpoint systems for music prediction. *Journal of New Music Research*, 24(1):51–73, 1995.
- D. Cope. *Computers and Musical Style*. A-R Editions, 1991.
- P. Cruz-Alcázar and E. Vidal-Ruiz. A study of grammatical inference algorithms in automatic music composition and musical style recognition. In *Proc. ICML-97 Workshop on Automata induction, grammatical inference, and language acquisition*, 1997.
- S. Dubnov, G. Assayag, and R. El-Yaniv. Universal classification applied to musical sequences. In *Proceedings of the International Computer Music Conference*, pages 322–340, 1998.
- M. Farbood and B. Schoner. Analysis and synthesis of Palestrina-style counterpoint using Markov chains. In *Proceedings of the International Computer Music Conference*, pages 471–474, Havana, Cuba, 2001. International Computer Music Association.
- L. Hiller and L. M. Isaacson. *Experimental Music*. McGraw-Hill, 1959.
- F. Jelinek. *Statistical Methods for Speech Recognition*. MIT Press, 1997.
- V. Jiménez, A. Marzal, and J. Monné. A comparison of two exact algorithms for finding the N -best sentence hypotheses in continuous speech recognition. In *Proc. 4th EUROSPEECH*, pages 1071–1074, 1995.
- D. Lidov and J. Gabura. A melody writing algorithm using a formal language model. *Computer Studies in the Humanities*, 4(3-4):138–148, 1973.
- L. B. Meyer. *Emotion and Meaning in Music*. University of Chicago Press, 1956.
- M. Mozer. Neural network music composition by prediction. *Connection Science*, 6(2-3):247–280, 1994.

- F. Pachet. Interacting with a musical learning system: the Continuator. In C. Anagnostopoulou, M. Ferrand, and A. Smaill, editors, *Music and Artificial Intelligence: Lecture Notes in Artificial Intelligence 2445*, pages 119–132. Springer-Verlag, 2002.
- M. Pearce. Improving the prediction performance of PPM variants with monophonic music. In *Proc AISB 2003 Symposium on AI and Creativity in the Arts and Sciences*, 2003.
- M. Pearce and G. Wiggins. Towards a framework for the evaluation of machine compositions. In *Proc AISB 2001 Symposium on AI and Creativity in the Arts and Sciences*, pages 22–32, 2001.
- R. Pinkerton. Information theory and melody. *Scientific American*, 194:77–86, 1956.
- D. Ponsford, G. Wiggins, and C. Mellish. Statistical learning of harmonic movement. *Journal of New Music Research*, 28(2), 1999.
- B. Reis. Simulating music learning: on-line perceptually guided pattern induction of context models for multiple-horizon prediction of melodies. In *Proc AISB 1999 Symposium on Musical Creativity*, pages 58–63, 1999.
- P-Y. Rolland and J-G. Ganascia. Musical pattern extraction and similarity assessment. In E. Miranda, editor, *Readings in Music and Artificial Intelligence*, chapter 7, pages 115–144. Harwood Academic Publishers, 2000.
- R. Rosenfeld, S. Chen, and X. Zhu. Whole-sentence exponential language models: a vehicle for linguistic-statistical integration. *Computers, Speech and Language*, 15(1), 2001.
- N. Ruwet. Méthodes d'analyse en musicologie. *Revue belge de musicologie*, 20:65–90, 1966.
- T. Sawada and K Satoh. Composer classification based on patterns of short note sequences. In *Proc AAAI-2000 Workshop on AI and Music*, pages 24–27, 2000.
- J. Sundberg and B. Lindblom. Generative theories in language and music descriptions. In P. Howell, R. West, and I. Cross, editors, *Representing musical structure*, pages 245–272. Academic Press, 1991.
- J. Triviño-Rodriguez and R. Morales-Bueno. Using multiattribute prediction suffix graphs to predict and generate music. *Computer Music Journal*, 25(3), 2001.
- M. Westhead and A. Smaill. Automatic characterisation of musical style. In *Music Education: An AI Approach*, pages 157–170. Springer-Verlag, 1993.

Symbolic Pointillism: Computer Art motivated by Human Perception

Norbert Krüger*

*Department of Psychology
University of Stirling
Scotland, UK
norbert@cn.stir.ac.uk

Florentin Wörgötter†

†Department of Psychology
University of Stirling
Scotland, UK
worgott@cn.stir.ac.uk

Abstract

A new kind of multi-modal image representation that results in computer generated images is introduced. This image representation has been motivated by human visual perception. We call the style of images ‘Symbolic Pointillism’ since it resembles certain impressionist images drawn by the sub-group of Pointillists. However, instead of points, symbolic icons represent condensed and meaningful local information.

1 Introduction

We introduce a new kind of multi-modal image representation that results in computer generated images with considerable aesthetic value (see figure 1, 4 and 5). This image representation is used in the artificial visual system described in (Krüger et al., 2002a,b; Krüger and Wörgötter, 2002; Krüger et al., 2003a; ModIP, 1998—) and is a central pillar of the ongoing European project ECOVISION (ECOVISION; 2002–2004). This image representation has been motivated by human visual perception (Krüger et al., 2003b). We call the style of images ‘Symbolic Pointillism’ as will be motivated below.

Pointillism is a style of art that was part of the impressionist movement. This movement aimed to paint images not in a realistic way but in a way ‘humans perceive the world’. The sub-group of impressionists called Pointillists (most notably Seurat and Pissarro) decided to base their paintings on small coloured dots (or even small oriented patches). The observer then constructs the image by mixing these dots. Perceived reality is, thus, a concept constructed by the observer.

It is by now known that the human visual system operated on a similar level. First, condensed and meaningful image information (like orientation, colour, distance, form, shape) is generated by localised processors (neurons) (Hubel and Wiesel, 1979; Gazzaniga, 1995). These neurons are highly connected and communicate with each other (Gazzaniga, 1995; Watt and Phillips, 2000). The need to reduce the cost of this communication process drives the condensation to meaningful information in the first place. As the consequence, it is almost as those nerve cells would in the end communicate symbols. It seems to be this specific neuronal communication process by which our perceived ‘reality’ is generated.

Having the aim of building technical systems with similar power and similar structure than the human visual system, we invented an image representation that consists of such abstract, symbolic icons. We include, for example,

information about colour, orientation and contrast transition (see figure 2). A communication process is established by grouping related information (as represented by linking lines). In this way our electronically generated paintings visualise internal principles of image processing in the brain.

In section 2, we give a short description of the computation of our image representation as well as its relation to human perception. We then discuss this approach in the context of the modelling of ‘creativity’ in section 3. As a central thesis, we suggest that the understanding and modelling of creativity can be supported by the understanding and modelling of human perception.

2 Computation of ‘Symbolic Pointillist’ Images

In section 2.1, we describe the processing of the basic entities of our images which are symbol-like icons. These icons carry condensed information about multiple aspects of local image structures and organise themselves into groups. In section 2.2, we discuss the specific role of the icons in visual processing.

2.1 Symbolic Icons and Early Visual Feature Extraction

In our artificial system we process an image representation that codes different feature domains.

Position and Orientation: In our icons (see figure 2a) orientation is represented as a local line with appropriate angle. Edge detection and orientation estimation is based on the isotropic linear filter (called monogenic signal (Felsberg and Sommer, 2001)). The monogenic signal performs a *split of identity*: it orthogonally divides a signal into energetic information (indicating the likelihood of the presence of a structure), its geometric information

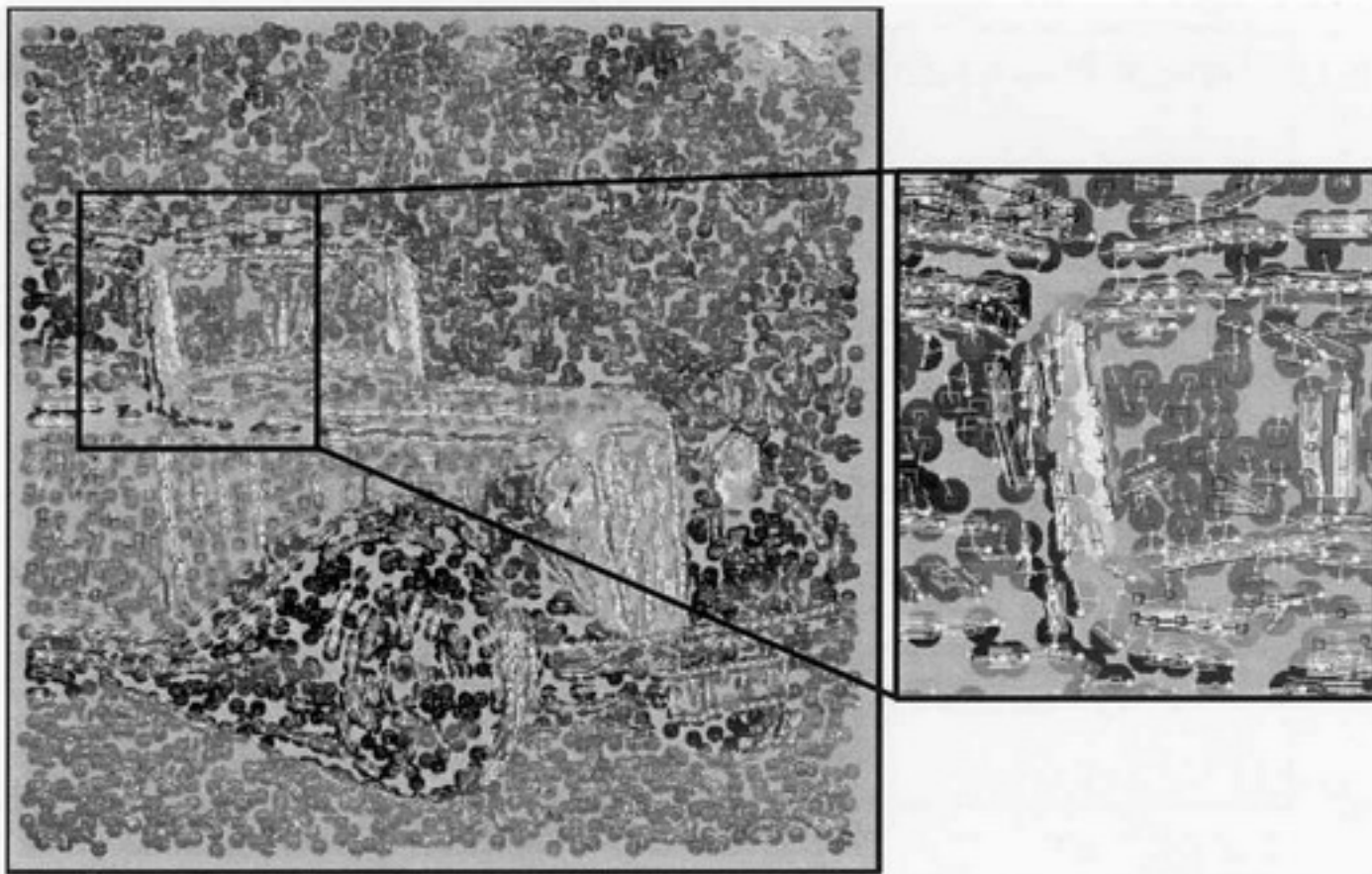


Figure 1: Symbolic Pointillism: A computer generated image and an enlarged frame.

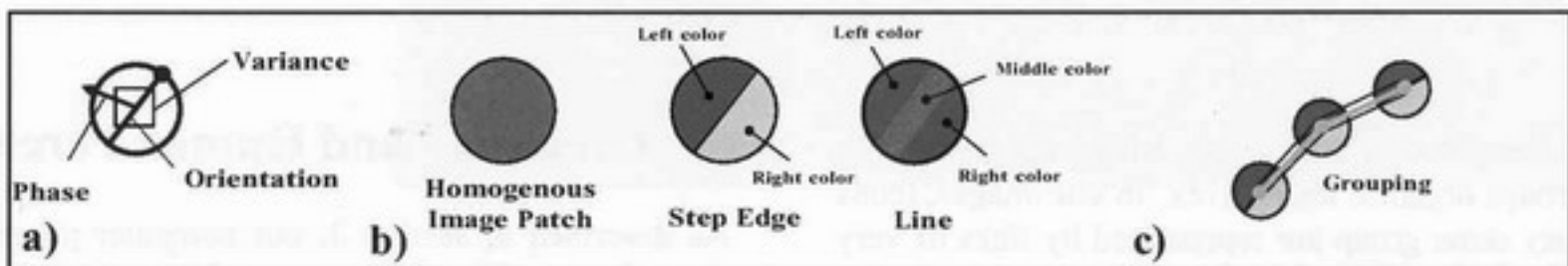


Figure 2: a,b: Icons as basic elements of 'Symbolic Pointillism'. c: Grouping is represented by linked lines.

(orientation) and its contrast transition. We look for energy maxima in the position-orientation space. We use hexagonally arranged patches with a diameter of approximately 15 pixels. To avoid the occurrence of very close line-segments produced by the same image structure we demand that line segments have a certain minimal distance.

Contrast Transition: The contrast transition is displayed by a small arrow (see figure 2a). Using contrast transition, we can for example distinguish between lines and step edges (see figure 2b). Contrast transition is coded in the phase at a local maximum in the (x, y, o) feature space (Kovesi, 1999). It refers to the kind of grey level structure existent at the local image patch (as dark/bright edge, or bright line on dark background). The continuum of contrast transition can be expressed by the continuum of phases. Therefore, it allows for a coding different kinds of edge-like structures by one parameter.

Colour: Colour is processed by integrating over image patches in coincidence with their edge structure. In case of a line we have in addition a colour value in a middle strip that carries colour information (see figure 2b). To code the modality color at intrinsically one-dimensional image structures we perform a Gaussian integration in the RGB color space over the left and right part ('left' and

'right' defined by the associated line segment) of the image patch (see figure 2). Since the distribution of phases indicates the dominance of edges (Krüger and Wörgötter, 2002), this kind of integration corresponds to the most likely model of intrinsically one-dimensional structures.

Energy and Orientation Variance: The 'homogeneous-ness', 'edge-ness' or 'junction-ness' of a local signal patch can be computed from the variance of the local orientation and the energy in this area. The variance is displayed as the diameter of a square (see figure 2a) while the local energy is displayed as the grey value of the upper part of this square.

There is a huge amount of evidence that the above-mentioned modalities are processed in early stages of visual processing in so called 'hyper-columns' (see, e.g., (Hubel and Wiesel, 1979; Jones and Palmer, 1987; Gazzaniga, 1995)). However, it is not only a local image processing that is going on in early visual processing. As mentioned above, there occurs an extensive communication within visual brain areas as well as across these areas. The communication process leads to a binding to groups (v.d.Malsburg, 1981; Watt and Phillips, 2000) of local entities. In (Krüger and Wörgötter, 2002), we have described a process in which such a binding develop based on statistical measurements in natural scenes. These sta-

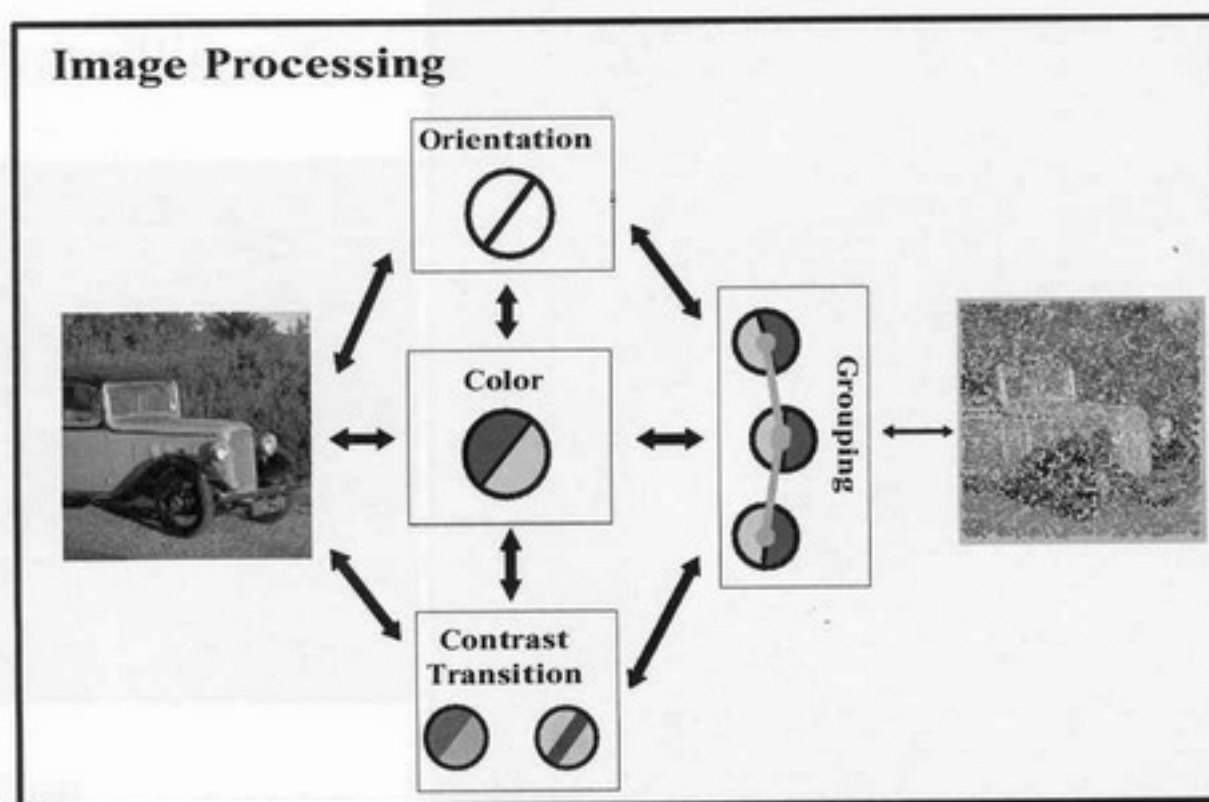


Figure 3: Different stages of image processing: First basic features in different domains (orientation, color, contrast transition) are being processed which are then grouped. Feature processing in the different domains as well as grouping is closely intertwined.

tistical measurements start a self-emergence process in which groups organise themselves. In our images, icons of the very same group are represented by links of very same colour (see figure 2c). The complete image processing is schematically displayed in figure 3.

2.2 Symbols as Carrier of condensed Information

All the low level features described in section 2.1 face the problem of an extremely high degree of vagueness and uncertainty (Aloimonos and Shulman, 1989). However, the human visual systems acquires visual representations which allow actions with high precision and certainty within the 3D world under rather uncontrolled conditions. The human visual system can achieve the needed certainty and completeness by integrating visual information (see, e.g., (Hoffman, 1980)) that occurs for example in the displayed grouping processes.

However, integration of information makes it necessary that local feature extraction is subject to modification by contextual influences and this communication has necessarily to be paid for with a certain cost. This cost can be reduced by limiting the amount of information transferred from one place to the other, i.e. by reducing the bandwidth. Therefore our symbolic icons represent a *condensed* description of a local image patch, which however contains the relevant information: Although a usual image patch has a dimension of, e.g., $15 \times 15 = 225$ pixel values, the output of our symbolic icons has less than 10 values.

3 Creativity and Human Perception

As described in section 2, our computer program produces images that display a certain amount of aesthetic value. People tend to find the images appealing, especially when they are displayed in a large format. Three images have been displayed (with a size of 1m^2 each) in 'The Lighthouse', a well known museum for modern art and design in Glasgow. A larger exhibition is on the way. Some images have been sold to a public institution. So they carry also financial value. But possesses our computer program creativity?

There is at least one aspect that may lead to a negative judgement plausible: Our computer program 'paints' natural objects and does not create 'paintings' using its 'imagination'. However, it has some understanding of visual structures in general and this understanding is displayed in the images by the design of the icons and their grouping. Therefore, the generated 'painting' is not a reproduction of nature but a significant *restructuring* of the perceived image. In this way, our computer program shows *its own style*.

We see the ability to be confronted, to process and to a certain amount understand visual input as an important property which is linked to creativity: It was this confrontation with the natural world that has been dominated the work of most artists and even in most abstract paintings general rules about aesthetic are applied that (with high likelihood) have their origin in certain statistical or geometric regularities in natural scenes (Mumford and Gidas, 2001; Krüger, 1998).

AARON, developed by Harold Cohen, was the first

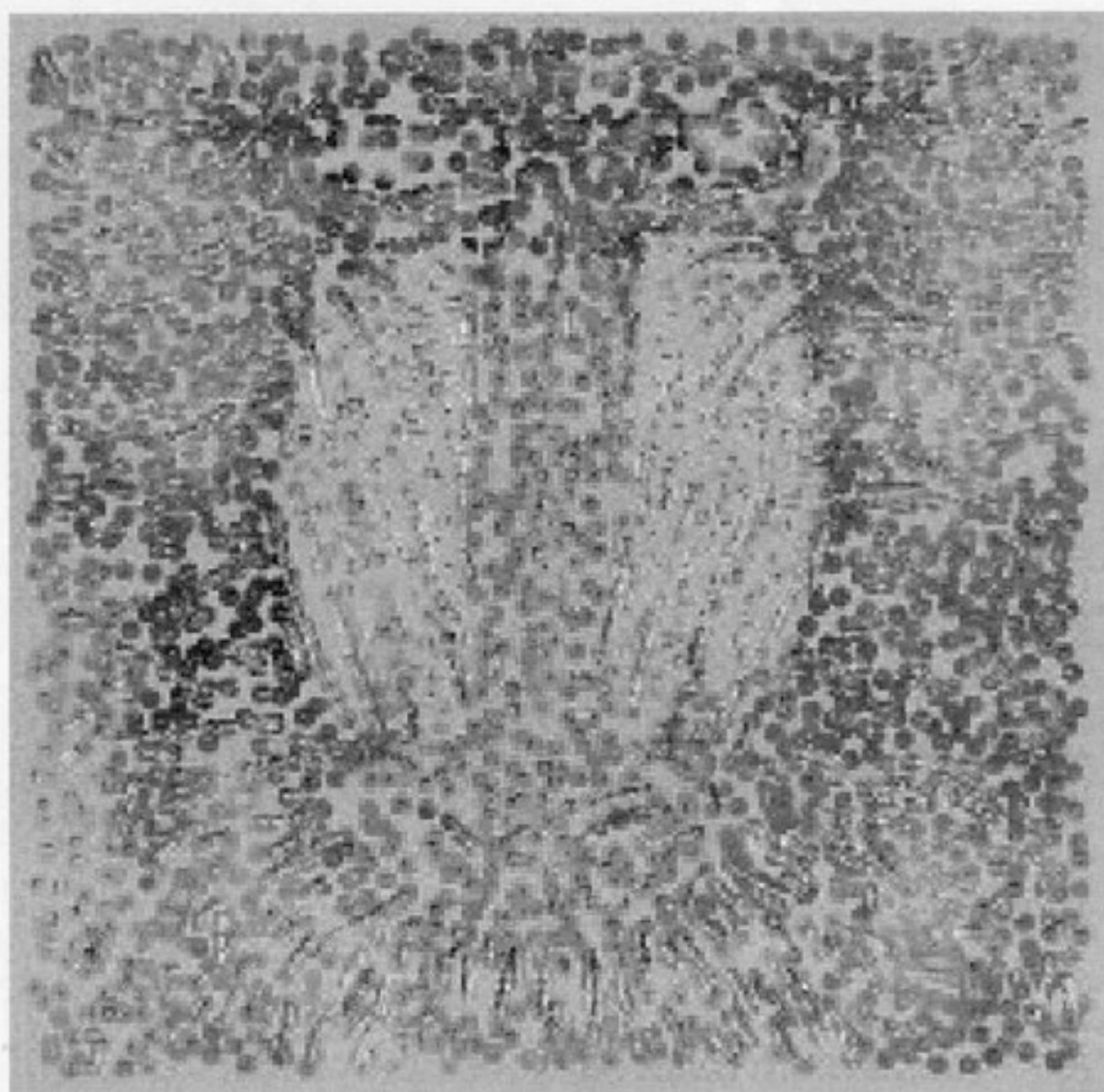
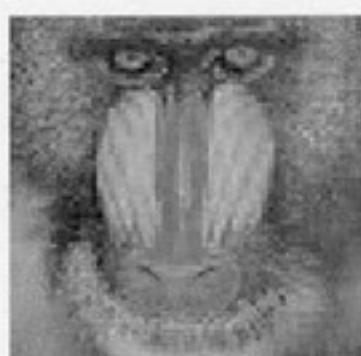


Figure 4: Symbolic Pointillism: Ape.

example of an ‘artificial artist’ that is able to paint images with considerable aesthetic value. Harold Cohen never claimed AARON to be creative. In contrast to our system, AARON was not equipped with the ability to perceive the world. It may well be that future artificial artists make experience in natural environments and develop their work in a process of similar confrontation with nature than many artists do. It may even be that the rules on which exploratory-creativity and transformational-creativity (Boden, 1990) is based upon, need to be learned from rules valid in natural environments. And it is the use, modification and deliberate violation of such well established rules that makes successful artists. In this way, we see our image representation as well as the produced images not only as an accidental ‘by-product’ of computer vision research but as a step towards artificial artists that are creative, and in this sense, can go beyond AARON.

References

- J. Aloimonos and D. Shulman. *Integration of Visual Modules — An extension of the Marr Paradigm*. Academic Press, London, 1989.
- M. Boden. *The Creative Mind: Myths and Mechanisms*. Aba-

cos, 1990.

- ECOVISION: Artificial visual systems based on early-cognitive processing, EU-Project. <http://www.pspc.dibe.unige.it/ecovision/project.html>, 2002–2004.
- M. Felsberg and G. Sommer. A new extension of linear signal processing for estimating local properties and detecting features. *Proceedings of the DAGM 2000*, pages 195–202, 2000.
- M. Felsberg and G. Sommer. The monogenic signal. *IEEE Transactions on Signal Processing*, 41(12), 2001.
- M.S. Gazzaniga. *The Cognitive Neuroscience*. MIT Press, 1995.
- D.D. Hoffman, editor. *Visual Intelligence: How we create what we see*. W.W. Norton and Company, 1980.
- D.H. Hubel and T.N. Wiesel. Brain mechanisms of vision. *Scientific American*, 241:130–144, 1979.
- J.P. Jones and L.A. Palmer. An evaluation of the two dimensional Gabor filter model of simple receptive fields in striate cortex. *Journal of Neurophysiology*, 58(6):1223–1258, 1987.
- P. Kovess. Image features from phase congruency. *Videre: Journal of Computer Vision Research*, 1(3):1–26, 1999.

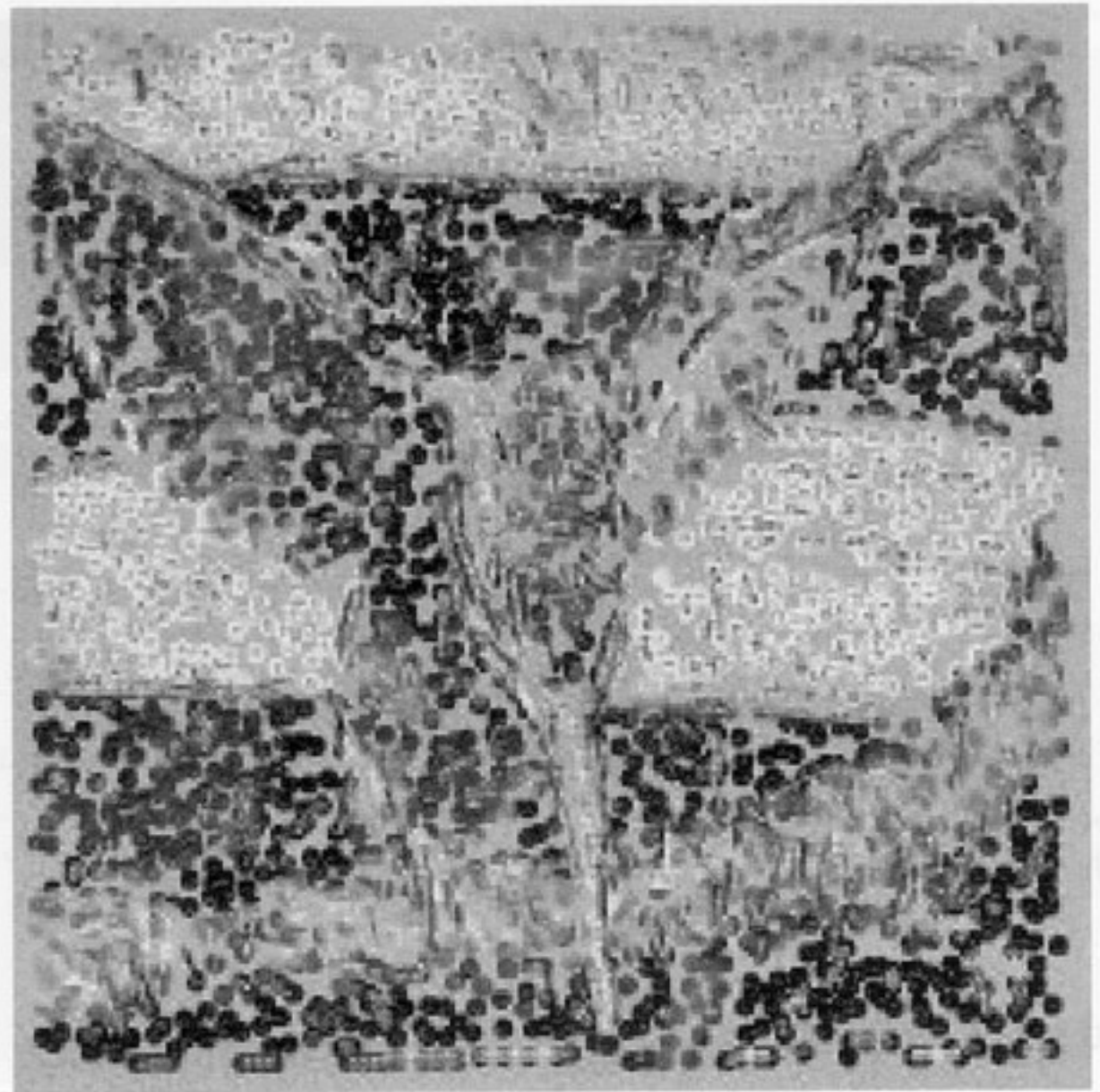


Figure 5: Symbolic Pointillism: Bird.

N. Krüger. Collinearity and parallelism are statistically significant second order relations of complex cell responses. *Neural Processing Letters*, 8(2):117–129, 1998.

N. Krüger, M. Ackermann, and G. Sommer. Accumulation of object representations utilizing interaction of robot action and perception. *Knowledge Based Systems*, 15:111–118, 2002a.

N. Krüger, M. Felsberg, C. Gebken, and M. Pörksen. An explicit and compact coding of geometric and structural information applied to stereo processing. *Proceedings of the workshop 'Vision, Modeling and VISUALIZATION 2002'*, 2002b.

N. Krüger, M. Felsberg, and F. Wörgötter. Processing multi-modal primitives from image sequences. *submitted to ICIP 2003*, 2003a.

N. Krüger, M. Lappe, and F. Wörgötter. Biologically motivated multi-modal processing of visual primitives. *accepted for 'AISB 2003 Convention: Cognition in Machines and Animals', Wales*, 2003b.

N. Krüger and F. Wörgötter. Multi modal estimation of collinearity and parallelism in natural image sequences. *Network: Computation in Neural Systems*, 13:553–576, 2002.

ModIP. (Modality Integration Project). <http://www.cn.stir.ac.uk/ComputerVision/Software/index.html>, 1998—.

David Mumford and Basilis Gidas. Stochastic models for generic images. *Quarterly of Applied Mathematics*, LIV(1): 85–111, 2001. ISSN 0033-569X.

C. v.d.Malsburg. *The correlation theory of brain function (internal report)*. 1981.

R.J. Watt and W.A. Phillips. The function of dynamic grouping in vision. *Trends in Cognitive Sciences*, 4(12):447–154, 2000.

Swarm Music: Improvised Music with Multi-Swarms

T. M. Blackwell

Department of Computer Science
University College London
Gower Street
London, UK
tim.blackwell@ieee.org

Abstract

This paper outlines Swarm Music, an improvisational musical system based on the dynamics of insect swarms. It is proposed that the interactions amongst performers in a freely improvising ensemble can be modelled using the self-organisational abilities of swarms. In a multi-swarm, each swarm is a musical entity which interacts with a target group. Each swarm is comprised of particles and each particle corresponds to a musical event. The targets are derived from other swarms, or from collaborating humans. The indirect interaction between swarms and humans is an example of stigmergy and can be expected to lead to temporal organisation. It is also proposed that the self-organisation of particles within a swarm leads to pleasing note patterns, so that each swarm is itself an interesting musical source.

1 Introduction

Real-time interactive music making with a machine is surprisingly rare. For example, a recent review finds only four systems that provide real-time music production (Miranda, 2001). Even rarer are systems that play improvised music – George Lewis' Voyager system is the most prominent (Lewis, 1999). Much effort has been spent on systems that only interact weakly with humans (for example, accompaniment programs that adjust pre-prepared material according to external parameters), or music composition programs, that embody some encoding of musical knowledge (as a grammar, for example) but do not produce music in real time. But the chief characteristic of improvised music is that it happens in real time and without revision. In fact interaction and improvisation have a symbiotic relationship, and this is to be heard most clearly in 'free improvisation'.

Freely improvised music (which differs from form-based improvisation as exemplified by jazz), is so named because it is free from many conventional musical constraints. In particular, no prior agreements are made on form or structure. Any temporal structure can only arise from the local interactions of the participants. In other words, it must be *self-organising*.

Examples of freely improvised music are to be found in the free jazz works of John Coltrane, Sun Ra and Cecil Taylor and, within the classical tradition, by composers such as John Cage, Cornelius Cardew and Karlheinz Stockhausen (this list is far from

complete). A wide ranging review is given by Dean (1992).

Since improvised music operates without any prior agreement on tonality, pulse or dynamics, the question arises as to *what* the musicians actually respond to. One answer is that the musicians are attentive to the *expressive* qualities of their musical environment. Expressive qualities are high level descriptions of the music, including attributes such as event density, average loudness and pitch. The participants interact by either trying to match the expressive quality of the musical environment, as they perceive it, or by attempting to change it in some way.

One novel approach for the design of an artificial improviser is to abandon schemes with in-built musical rules, and to look towards nature for systems that are self-organising. Then, a mapping must be made from the extra-musical system onto actual musical events. This mapping would not encode musical syntax; musicality should be an emergent property (Blackwell, 2001, Blackwell and Bentley, 2002b). The collective behaviour of social animals offers a tempting analogy.

Insect swarms and avian flocks are examples of natural systems that exhibit self-organisation (Bonabeau, 1999) and seem to be analogous to improvised music. For example, a flock has a persistent shape and it may suddenly change its direction of flight, with every member almost spontaneously responding. Flocks and swarms may split, or they may join. Swarm members are 'attracted' towards the swarm as a whole, yet avoid

colliding with their neighbours. This remarkable behaviour appears to be choreographed or led. In fact, merely local interactions between adjacent flock members are sufficient explanation (Reynolds, 1987).

The same de-centralised patterns are also evident in improvised music. Despite the lack of any pre-arranged form or material, improvisations can produce remarkable structures. For example, sudden changes in musical direction can arise whereby every member of the improvising ensembles departs on a new trajectory, as if orchestrated and conducted. And at a different level, the pattern of notes arising from the contributions of an individual may be swarm-like. The succession of notes may move around a key centre with a characteristic 'shape' – the pattern of rising and falling tones. Melodic variations can be visualised as gradual changes in shape. Notes may avoid 'collisions'. Similarly, the placement of notes in time gives patterns. There may be attraction to a beat, resulting in a simple organised pattern. Often, though, rhythms are a pattern of accents and placements about a beat, implying a degree of 'collision avoidance'. Chords are groups of notes with an avoidance of small intervals and an attraction to consonant intervals. Good harmony follows the rules of voice leading – notes make the minimum movement necessary for the harmonic progression. Chord progressions are 'attracted' to a key centre, but they may actually avoid the tonic for some time, delaying resolution.

The idea underlying Swarm Music is that the musical improvisations of an individual can be generated by a swarm of musical events. Individuals then interact with one another by expressive couplings. This interaction is realised by groups of attracting targets. Each target in a group is a captured event from another swarm, or from an external source. The swarm moves towards the targets, outputting musical events. These events are captured by other swarms, and become targets in their own target group. Here the term multi-swarm will refer to a colony of swarms with indirect target mediated interactions.

Previously published work on Swarm Music (Blackwell, 2001, Blackwell and Bentley, 2002b) outlines a single swarm system with simple animation. This paper extends that work by considering multi-swarms. The animation algorithm has been extended and the relationship of the parameters to the musical output is now better understood. Other improvements noted here are in event parsing and target placement.

Although Swarm Music may operate autonomously, it may also be directed by a process analogous to the conducted improvisations practised by the musicians Butch Morris and John Zorn and known as 'conduction'. In this approach, a vocabulary of gestures influences but does not specify the improvisations of a group of musicians.

Recent work on conducted swarms is also described here.

This paper continues with an account of some of the theoretical principles underlying Swarm Music. The next section fills in some details and describes the three constituent modules in each swarming system of the multi-swarm. Then, the generation of a number of improvisations, is discussed in some detail.

The aim of Swarm Music is both to show that the dynamics within an improvising group can be modelled by interacting swarms, and to develop a practical application. In fact the two aims coincide, since a convincing performance by any such application would support the hypothesis. With this in mind, an attempt to evaluate the system from the perspective of the end user is given in section V. The paper closes with a discussion of the system and its place within the wider context of computer creativity.

2 Background

2.1 Self-organisation and stigmergy

The self-organisation of a system is the appearance of spatiotemporal structure due to low level interactions amongst its constituents (Bonabeau, 1999). There are believed to be four contributing mechanisms: positive feedback, negative feedback, amplifications of fluctuations and multiple interactions. For example, the pheromone trails of foraging ants are self-organised. The laying of a trail by an ant and the attraction of other ants to this trail is a positive feedback, but the evaporation of established trails is a stabilising negative feedback. Ants also wander randomly and may discover a new food source. This can then be amplified by positive feedback. Multiple interactions exist amongst the ants; they can make use of the results of their own activities and of the colony as a whole (i.e. they can follow their own trails, or trails left by other ants).

The interactions between individuals in a self-organising system may be direct or indirect. Direct interactions include actual contact between individuals, or contact mediated by sound, smell or vision. Indirect interactions occur when an individual modifies the environment, which other individuals respond to at a later time; this form of communication is known as stigmergy (Bonabeau, 1999).

2.2 Particle swarms

The animation of each swarm in Swarm Music is inspired by the classic flock animations of Reynolds (1987). In these animations, the individuals or boids are small objects with flapping wings, an orientation, a position and a velocity. Each boid experiences a number of accelerations. An attractive acceleration

draws boids together, and a velocity matching acceleration encourages parallel motion. A third acceleration prevents collisions and enables the flock to fly around obstacles.

In a particle swarm, the finite-sized boids are replaced by point particles, and the velocity matching term is removed. In particle swarm optimisation, the attraction of a particle towards a local neighbourhood is replaced by attractions towards good solutions found by the particle and by the swarm as a whole (Kennedy and Eberhart, 1995). The particle swarm in Swarm Music, however, retains particle-swarm attraction, but introduces the concept of ‘charge’ (Blackwell and Bentley, 2002c). Charged particles experience Coulomb repulsion; this is collision avoidance and serves to enhance the diversity of the swarm, a useful property for dynamic optimisation.

2.3 Swarm Intelligence

Self organisation enables a colony to solve many problems (e.g. discovery of food sources, building nests). Swarm intelligent systems are directly inspired by this problem solving ability. Two recent publications demonstrate the use of ant colony and particle swarm algorithms in optimisation problems (Bonabeau, 1999, Kennedy and Eberhart, 2001).

Swarm Music uses self-organising particle swarms. Particles within the multi-swarm interact directly with fellow members from their own swarm, and indirectly with particles from other swarms via targets in their own target group. The size of the target group is finite and limited to the corresponding swarm size. This means that targets will be replaced as new events are captured – they have a finite life time. Moreover, the target groups will represent captured events from any swarm or external source, depending on the order of capture. The result is a stigmergetic relationship between the swarms. Positive feedback is present because targets are attractors for all particles, and because all particles in a swarm feel a mutual attraction towards the swarm centre. However negative feedback is also included in particle-particle and target-particle collision-avoidance accelerations. This negative feedback operates between particles in the same swarm and has the effect of introducing disturbances and maintaining the spatial distribution of the swarm (Blackwell and Bentley, 2002a). The disturbances are then amplified by stigmergy.

3 System Overview

Swarm Music has a number of independent systems, each responsible for one swarm in the multi-swarm. Each system has three functions: capture, animation and interpretation. The animator updates the positions of each particle. The swarm is then interpreted by mapping positions onto musical events in music space. Each event is then played on the internal

synthesizer and visually displayed as a 3D animation. Concurrently with animation/interpretation, external events and events originating from other swarms are captured, parsed and placed as targets in the physical space of the swarm. There are a number of adjustable parameters for each function. These may be changed in real-time (conduction) or just left fixed (autonomous).

3.1 Animation

Each swarm is made up of particles, and each particle has a position and a velocity in a Euclidean or ‘physical’ space. Each particle is subject to one or more of four accelerations. These accelerations are summed and added to the velocity. This updated velocity is then added to the particle position in order to advance the particle by one time step. The accelerations are indexed by k and described in Table 1.

Table 1: Particle Accelerations

k	Description
1	Attractive acceleration towards the swarm centre of mass.
2	Attractive acceleration towards the target group centre of mass.
3	Attractive acceleration towards a particular target.
4	Repulsive acceleration away from a neighbouring particle or from a particular target.

Each acceleration is a power-law of the form $a_k = C_k |r|^{n_k - 1} r$ where r is the separation vector to a particle or target and n_k is the associated index. Negative values of C_k correspond to attraction, positive values to repulsion. For the case of negative index, the acceleration is set to a constant for separations smaller than a small value (core radius), in order to remove the singularity at $r = 0$. Additionally, a limit of perception is set so that so that particles do not experience repulsions from other particles or targets at separations beyond this limit.

The first version of Swarm Music (Blackwell, 2001) used spring-like accelerations ($n_{1,2} = 1$) towards the swarm and target swarm centres of mass and an inverse square inter-particle repulsion, $n_4 = -2.0$. Acceleration parameter C_3 was set to zero so that there was no attraction to *individual* targets.

Subsequently, different acceleration indices were tried, but the musical results did not vary much (whereas the indices would be more crucial for optimisation purposes). The important factors, from a musical perspective, are how much attraction and repulsion is present (determined by the acceleration constants) and which particles serve as attractors. Swarm Music currently implements all four

accelerations, and allows the user to specify combinations and strengths.

The accelerations balance individual and group behaviour. For example, particles are attracted towards the swarm centre, which preserves the identity of the swarm as a whole, yet may also feel attractions to targets that lie outside the swarm, tending to fragment the swarm. Also, the inter-particle repulsion (which is only appreciable at small distances) balances the attraction towards the swarm centre, preventing collapse. Musically, this prevents the swarm from clumping together, thereby playing identical notes.

Table 2 specifies the main animation loop in detail.

Table 2: Main Animation Loop

k	Description
1	Attractive acceleration towards the swarm centre of mass.
2	Attractive acceleration towards the target group centre of mass.
3	Attractive acceleration towards a particular target.
4	Repulsive acceleration away from a neighbouring particle or from a particular target.

Here, each particle i , $i = 1, 2 \dots M$ has a 3-dimensional position x_i and velocity v_i . The targets have a position y_i and are distributed in the target cube T of extent x_{max} in each dimension. A single swarm update or iteration now involves an update of each particle in turn. Each spatial dimension $a = 1, 2, 3$ has an associated clamping velocity $v_{max, a}$. No velocity component $v_i \cdot e_a$ (e_a is the associated unit vector) is allowed to exceed this speed limit. Particles that do stray outside T are reflected back.

The **update parameters** are those specified in the main animation loop. The acceleration parameters C_k , limit of perception and core radius can be adjusted for each swarm. In addition, each of the four accelerations can be switched on and off. The components of v_{max} are adjustable between zero and an arbitrary maximum. If $|v_{max}|$ is small in each dimension, the swarm will only vary slightly in shape from update to update. The result is a slowly evolving riff. Alternatively $v_{max, pulse}$ may be small and $v_{max, pitch}$ and $v_{max, loudness}$ large. The swarm will then respond quickly to new events in the pitch and loudness dimension, but slowly to new pulses. New melodies will be generated, but each melody will share a rhythmic pattern. Finally, v_{max} can be automatically scaled according to particle-target separation so that the particles slow down as they approach a target.

3.2 Interpretation

Swarm interpretation is directly analogous to the interpretation of a score. A simple interpretation must be found that is musical and which does not presuppose too much. The motivation and details of interpretation have been described by Blackwell and Bentley (2001, 2002b). A brief summary will be given here.

Figure 1 shows a view of the swarm in physical space.

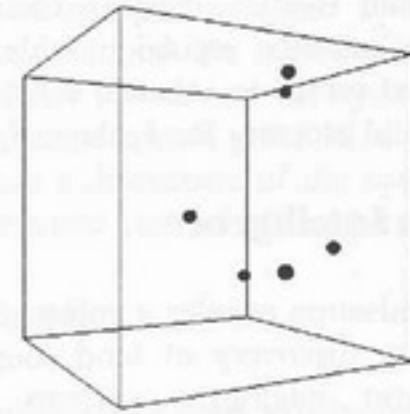


Figure 1: A view of the Swarm

Interpretation is a mapping from this space to the space of musical events – Music Space. Music space is populated by musical events, each one of which corresponds to a note played at a certain time and with a definite loudness. The three axes of music space are therefore loudness, pulse and pitch, which equates to the expressive parameters mentioned above (Blackwell, 2001).

Figure 2 shows a single event in music space.

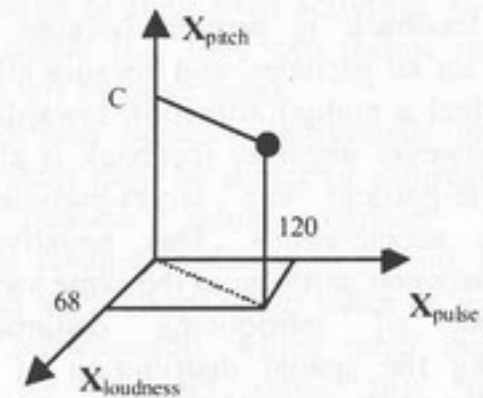


Figure 2: An event in Music Space

The event corresponds to middle C (C3), played at loudness MIDI 68, and sounding at a time interval corresponding to a beat of 120BPM after the preceding event. The mapping itself depends on the settings of the style parameters (see below) and is linear in loudness and pitch. The pulse mapping is more complicated and involves interpreting the physical separation along a given axis as a time interval Δt between notes. This is inverted to give a

rate (i.e. beats per minute) and stretched so that very high and low BPM's are suppressed.

The **style parameters** govern this interpretation and how each note is played by the synthesizer. The swarm can be played by ascending order of pitch, and each note can be sustained. The rhythm of the notes can be stabilised by taking the pulse from the swarm centre of mass, rather than from each particle. The size of T may be adjusted to restrict the interpretation of a swarm to, for example, a specified instrumental range. Finally, the choice of MIDI instrument can be specified and the interpretation can be locked into a particular key and scale.

3.3 Capture

Each system captures both internal MIDI events generated by other swarms and external MIDI and audio events, parses these events for pitch, pulse and loudness, and sets targets in the physical space of the swarm. (The audio stream is split into events by a pulse height analyser – pitch is determined with a real-time fast Fourier transform.) At this stage there is much scope for directing the improvisations since the placement of the targets provides the stimulus for particle motion. Although the process can be completely transparent, a number of **scripts** can also be implemented to give the user some measure of control.

For example, one script will randomise the target group if the swarm converges on the target centre of mass. This will happen if there is no stimulus from outside the swarm, or if the stimulus is constant. The effect is to give the swarm a new musical direction in response to either an absence of ideas from its collaborators, or from too much similarity. In terms of the biological analogy, this is equivalent to the full exploitation of a food source; the swarm then scatters to enhance diversity until a new source is found.

Another script parses captured notes for key and scale (which can be tonal or synthetic). Interpretation is then made within this key and scale. Key and scale changes are invoked in response to harmonic changes in the input. The script also tolerates 'mistakes', only instigating key or scale changes if the number of incorrect notes surpasses a certain number. This is important since, within the context of improvised music where no form has been previously agreed, the only way to know if a note is an error or a deliberate dissonance (in which case no action should be taken) or an implied change (in which case some type of action *could* be taken) is to see if the event is repeated. The effect of this simple algorithm is to produce fascinating chases from key to key, from moments of bi-tonality (with two swarms) to periods of consensus.

Other new scripts include improved target placement. Targets can be scaled so that, for example, a swarm playing in a low bass range will adjust incoming events by subtracting octaves so that

targets are placed in the appropriate pitch range. The responsiveness of a swarm to new target positions can now also be controlled by altering the target update time. Usually this is an almost continuous process, but the target update time can now be adjusted to longer intervals.

In another 'chordal' script, a number of particles can be allowed to sound at one time – part of the swarm is interpreted as a chord. This is governed by an adjustable parameter, Δt_{chord} , which is chosen by the operator; if n events are captured within Δt_{chord} , then n particles are sustained. The chord finishes probabilistically, where the 'notes-off' probability per iteration is also adjustable.

4 Recordings

A written analysis of musical improvisation is lengthy and difficult. However, two cds (Blackwell and Higgins 2001, Blackwell 2002) of Swarm Music improvisations with humans and of solo multi-swarm performances have been recorded and excerpts are available at <http://www.timblackwell.com>. Three of these excerpts are discussed below.

4.1 Follow Me (conducted improvisation)

This is an example of a conducted improvisation. Two three particle swarms were used. One of these swarms (A) plays sustained chords. At each update, three more notes are added to the growing chord. When the chord becomes too dense, the conductor kills the chord. Swarm B is playing a melody, mainly on a vibraphone, although the instrumentation does change during the piece. At certain places we can hear the swarm rapidly bouncing up and down in pitch. This was achieved by squeezing the pulse axis so that particles are interpreted within a narrow (and fast) range of beats per minute. At these times, and in pursuit of swarm A, swarm B began to oscillate with large amplitude in the pitch dimension. The 3D animation showed a gripping chase in music space. The behaviour was quite a surprise to the conductor!

4.2 Autumn Missed (autonomous)

Autumn Missed is a piano improvisation, performed by two five particle swarms - swarm A is playing left hand and swarm B is playing right hand. This was achieved by placing all targets below or above middle C for swarm A and B respectively. It is evident that both swarms are playing both melodies and chords. This was achieved by adjusting two sets of parameters. Firstly, MIDI "all notes off" messages are sent only at the start of each interpretation of the swarm (rather than after each particle has been interpreted) with probability one half. In other words, it is as if the sustain pedal is pressed down but momentarily released every five notes with

probability one half – except the piano has two sustain pedals, one for each half of the keyboard! The second parameter that is set to achieve the melody/chord effect is Δt_{chord} . This parameter is a time window for identifying chords. Since all MIDI events occur separately, the event parser needs to recognise chords as events arriving within a time interval. Normally this is set to a small value (5ms) so that chords can be clearly distinguished from very fast melodies. However, in this case, Δt_{chord} was set to 0.5 s. The event parser will force the interpreter to play interpreted notes as chords, where the number of notes in the chord corresponds to the number of notes in the captured chord. This setting will persist until two melody notes are heard, in which case all particles in the swarm will be interpreted as melody notes.

Despite the simplicity of these algorithms, when coupled with the self-organising properties of the two swarms, the effect is very musical, sounding very much like a piano improvisation of human origin.

4.3 Wind Up (autonomous ensemble with human)

This improvisation is an attempt to re-create a typical free jazz quartet with percussion, piano, bass and saxophone. In this recording a 3-swarm was used for the rhythm section (percussion, piano, bass) and a human supplied the saxophone improvisation. The bass can be heard to play a ‘walking’ line of steady tempo, only slowing towards the end of this excerpt. (This sense of pulse is typical of most of free jazz, which is not strictly freely improvised in the sense alluded to in the introduction. For example, much of the work of Ornette Coleman who is considered to be a founder of free jazz, retains a pulse and a song form, but is ‘free’ harmonically (Jost, 1994).) The percussion sounds were achieved by restricting the pitch dimension of one of the swarms to MIDI 35 – MIDI 52 and routing the output to MIDI channel 10, so that the each note corresponds to a percussion instrument on the standard GM drum kit. The bass and percussive swarms are well matched for pulse, but the piano-voiced swarm plays at a slower pulse with short melodies punctuated by chords. This was achieved with the chordal script described in section 3.3. The effect is certainly not unlike the ‘comp’ style of jazz accompaniment.

The correlation of activity within the multi-swarm is evident from the coherence of the roles of piano, bass and percussion throughout this excerpt. Particularly interesting from the perspective of self-organisation is a piano chord at 2:10 (min:s) near the end of the excerpt. At this point the pulse slows considerably from somewhere near 230 BPM to about 80 BPM. All three swarms take part in this temp change, possibly caused by the amplification of a fluctuation (as described in section 2.1). The saxophonist responds to the new direction.

The subtle response of the multi-swarm to the external input is harder to determine and requires careful listening. For example, two high notes from the saxophone at 1:20 prompt the bass line to rise at 1:25. Small crescendos and diminuendos involving the swarms and the saxophone are evident through the excerpt. In another pleasing episode, two multiphonic notes from the saxophone at 0:52 provoke another rising bass melody at 0:58 (targets being set from the higher harmonics of the captured multiphonic), and a flurry of saxophone notes is matched by a rapid beating from the percussion.

Overall, the excerpt does resemble free jazz, with evidence of interaction between the ensemble members, and a change in musical direction very characteristic of swarms.

5 Evaluation

5.1 Aims

The aim of Swarm Music is to produce a human-comparable improviser within the context of freely improvised music. This objective, if achieved, would indicate that freely improvised music is self-organising. The context is a ensemble improvisation, conducted improvisation or a solo improvisation. In conducted improvisation, the individual autonomy, as compared to an ensemble improvisation, is reduced since each member is expected to react to gestures emanating from the conductor. A solo improvisation which lacks interaction would have to be distinguished from a composition (also lacking in interaction) by the uncertainty of the musical output and the absence of any prepared musical structure.

5.2 Methodology

A modified Turing test is an obvious way of evaluating this aim. The test though is problematical since the listening experience of the subject is clearly very important. To the uninitiated (or even the most experienced) listener, free improvisation may always sound incoherent, in which case an artificial system would produce human comparable results just with a random output. On the other hand, a very experienced critic would begin to rate the performers, with the result that the artificial system would have to out-perform at least one of the humans if it were to pass. But it may be the case that the system is indeed human comparable but just not as good as the rest of the ensemble.

In the lack of such a test, a recent concert involving Swarm Music provides some indication as to how close Swarm Music is to fulfilling its objectives. Feedback from the musicians involved provides anecdotal evidence, and is prone to many problems of bias, but it can form the basis of a preliminary assessment.

5.3 Swarm Music in Performance

A concert, which was titled "Sonic Explosion", took place at Brunel University, UK, on February 26th 2003. Swarm Music was involved in three performances. The first was a duet with saxophonist Tim Whitehead who is a very experienced improviser. This was followed by a duet with Kathleen Willison, a professional jazz singer. The final piece was an ensemble improvisation with Whitehead, Willison, the composer Colin Riley (keyboards) and percussionist Robert Millet. The three pieces amounted to some 35 minutes of improvisation.

Both duettists had rehearsed with Swarm Music for 30 minutes prior to the concert, and interestingly chose different ways of working. Whitehead arranged three cues with the operator (me). Working with a single swarm, the style parameters were set so that the key and mode could not be changed by the swarm, but were only changed by the operator when cued to do so. This enabled Whitehead to force a change of key and mode. On the other hand, Willison wished to work with the set up of section 4.2 (Autumn Missed) - namely 2 autonomous swarms which can choose keys and modes from captured events. This results in periods of bitonality and introduces a degree of uncertainty since the pitch recognition algorithm can sometimes 'mishear', and identify a higher harmonic (Blackwell, 2001). However this can (and was) used to artistic effect by Willison.

It is important to comment on the involvement of the operator in the process, since the style parameters and scripts can be altered in real time, and whilst this does not amount to control of the multi-swarm, it clearly does have some influence. Again the improvisers had different preferences. Whitehead thought that any conduction could complicate his understanding of the swarm, so no adjustments were made to style parameters or to script. Willison was happy for there to be a degree of conduction, and the operator introduced the chordal script into swarm B from time to time, with occasional instrument changes.

The third ensemble improvisation was set up so that Swarm Music only received input from Willison. The ensemble could then react to the duet between Willison and the 2-swarm (and of course to each other). New musical directions initiated by the ensemble could therefore be indirectly passed to the multi-swarm via Willison. This configuration, which was chosen by Riley, is notable because he wanted to experiment with indirect as well as direct interactions between improvisers. This can be interpreted, from the point of view put forward in section 2, as stigmergy (I had not previously discussed this idea with him). In addition, the operator conducted the swarm, once more by switching the chordal script on

and off and by making adjustments to the limits of the pulse dimension of either swarm.

5.4 Results

The audience seemed happy enough with the performance, and directly afterwards I interviewed the two duettists.

Whitehead thought that Swarm Music was "more assertive and much easier to converse with" in the ensemble improvisation. He also commented that "...the lyricism was so high that I felt compelled to follow it, mimic it, reply to it..." However he found the standards sampled sounds in the sound module were limiting "...needs some really good sounds!"

Willison was surprised to find in the first improvisation that Swarm Music seemed to be imitating her "(the swarm) hit the same note at the same time - the harmonies worked". However, there was some tension; "At times I would have liked it to slow down... it has a mind of its own...give it some space". Her solution to the "forward motion" of the swarms was to "wait and allow the music to catch up". In general she thought that "performers must be prepared to be open minded (and) sure of your notes or (the music) could fall apart... (but you) don't need to be the best improviser, it will react to you." Overall she did seem happy with the performance - "(it) felt like we were making music!"

5.5 Assessment

The positive reaction of the audience and the performers to the three improvisations, and indeed the willingness for professional musicians to participate, demonstrates that Swarm Music is, at some level, successful. However this does not by itself show that the aims have been achieved.

The remarks are again encouraging but need to be assessed critically. For example, it may be argued that the observation by Whitehead concerning "assertiveness" in the third piece, suggests that the increased activity of the operator is important. However, alteration of the style parameters and scripts does not determine in detail the melodic, rhythmic and harmonic content of the output, or the self-organisation of the swarms around the targets. Rather, the operator sets boundary conditions, and parameter values which *influence* the swarms. In other words, if the swarm produced random unorganised musical material, alteration of style parameters and scripts would not ameliorate the output. Operator meddling could of course diminish the musical output, and this might have been a factor in Whitehead's request for no conduction during his duet! The quality of his own response whilst playing, and his later comments on lyricism, suggest that he found Swarm Music to be a human-comparable partner.

Willison also found the swarms to be musically intriguing. Her frustration at the occasional disobedience of the swarms is also typical of human improvising partners, and is often a source of creative tension. To be disobedient also implies the ability to be obedient, and her remarks show that she did feel there were times when she and the swarms were collaborating, eventually giving her the impression that they were "making music".

Ultimately, the judgement will be aesthetic. For example, it is not enough just to demonstrate that the multi-swarms respond: they must be shown to do so in a musically fresh way. Also, the music must have appeal beyond the experience of the performers; with this end in mind, it is hoped that a recording of Sonic Explosion will soon be available.

6 Discussion

The recordings and concert demonstrate an ability to participate in improvisations with humans or with other swarms. An exciting new development, made possible with multi-swarms, is conducted improvisation. The recordings demonstrate that the multi-swarm is self-organising. Indeed, the results are surprisingly musical; little intervention is needed to sustain an interesting improvisation. In fact the result is almost too intricate. Whitehead remarked during the rehearsal for Sonic Explosion that there are "too many ideas!".

The crucial update parameters are now better understood. The introduction of velocity clamping in each dimension is a very useful device (evident on some of the recordings from the cd "Autumn Missed" (Blackwell, 2002) but not described here). The attractive accelerations are spring-like, and this might be expected to give rise to oscillations in each dimension. However the interactions amongst particles and between swarms distort this regular motion. Even so, a remnant of the oscillatory motion does persist and it might be one reason for the musicality of the melodies (Blackwell, 2001 and Blackwell and Bentley 2002b).

All the style parameters are selected by the operator (although scripts can override them - see below) and can be changed at any time. The system can be extended by adding dimensions to the physical space so that swarming would be possible along directions corresponding to other musical attributes. The duration of each note Δt_{event} is an obvious candidate.

Sonic explosion demonstrated that Swarm Music may perform autonomously, or may be 'conducted'. It would be interesting to determine some heuristics that apply when an operator feels compelled to make adjustments to the style parameters and scripts. For example, if a random target script is switched on, then what prompted this decision? The music may have appeared boring to the conductor, but is

boredom an absolute quality? The adjustable parameters will allow a performer to program in a 'boredom threshold', and maybe even a dynamic boredom threshold adjuster - but what determines this? These are complex issues and Swarm Music does not operate at this level.

However it would be very interesting if swarming in these higher levels could take place. To begin, the higher levels would have to be quantified so that attraction could be defined. For example, the boredom threshold (convergence radius around the targets) could itself become a target. This target would be placed according to analysis of captured events. If captured events clump together in Music Space for a certain time before separating and then re-grouping, an estimate of the boredom threshold of the source can be made, and a target set.

Another possibility for higher level swarming would be in frequency of repetition. Since a swarm can be made to riff by setting v_{max} zero, a riffing frequency of captured material can be worked out, and targets set along this dimension. Similarly, the rate of phrase development is determined by moving v_{max} away from zero, which would be another target, if this quantity was estimable from the captured events.

As it stands, Swarm Music can only structure its output by the self-organisation of low level events. It may, by chance, discover a sonata or a sixteen bar form, but this is very unlikely. Swarming in the space of possible forms is a fascinating idea, but it is hard to see how parsing the incoming low level events can lead to target setting in this space.

Although Swarm Music has been mainly used in an improvisational context where the music is performed without revision, it could also be used as a compositional tool. The context for this is set out in a publication on Creative Evolutionary Systems (CES) (Bentley and Corne, 2002). Within CES, systems evolution is open-ended rather than goal orientated, and this can lead to creative problem-solving. These systems have been used by designers, artists and jazz musicians to provide novel and creative solutions.

The multi-swarm is also not intent on a specific goal (such as the optimisation of an external objective function). On the other hand it is not aimless; attracting targets deliver a purpose, analogous to the search for food in an actual swarm. A composer can experiment with scripts which govern this search, retaining any interesting patterns that the swarms produce.

This paper has outlined some of the recent development in the Swarm Music project. Swarm Music will continue to perform with improvisers at various events. Undoubtedly the experience gained from live performance, where unprepared music must be generated by a machine, and in interaction with humans, will lead to further modifications and insights.

References

- Bentley P.J. and Corne D.W. Creative Evolutionary Systems. *San Diego: Academic Press*, 2002
- Blackwell T.M. Making Music with Swarms. *MSc thesis, University College London*, 2001
- Blackwell T.M. Autumn Missed. *ownlabel cd 002, (available from author)*, 2002
- Blackwell T.M. and Bentley P.J. Don't Push Me! Collision-Avoiding Swarms. *Proceedings of the Congress on Evolutionary Computation Piscataway NJ:1691-1696*, 2002a
- Blackwell T.M. and Bentley P. Improvised Music with Swarms. *Proceedings of the Congress on Evolutionary Computation Piscataway NJ:1462-1468*, 2002b
- Blackwell T.M. and Bentley P. Dynamic search with Charged Swarms. *Genetic and Evolutionary Computation Conference, Morgan Kaufman, San Francisco:19-26*, 2002c
- Blackwell T.M. and Higgins R. Swarm Music. *ownlabel cd 001, (available from author)*, 2002
- Bonabeau E., Dorigo M. and Theraulax, G. Swarm Intelligence: From Natural to Artificial Systems. *Oxford University Press*, 1999
- Dean R. New Structures in Jazz and Improvised Music since 1960. *Open University Press*, 1992
- Jost E. Free Jazz. *Da Capo Press, New York*, 1994
- Kennedy J. and Eberhart, R.C. Particle Swarm Optimisation. *Proceedings of the IEEE International Conference on Neural Networks, IV:1942-1948*, 1995
- Kennedy J. and Eberhart R.C. Swarm Intelligence. *Morgan Kaufman*, 2001
- Lewis G. Interacting with Latter-day Musical Automata. *Contemporary Music Review* 18(3):99-112, 1999
- Miranda E.R. Composing Music with Computers. *Oxford: Focal Press*, 2001
- Reynolds C.W. Flocks, herds and schools: a distributed behavioural model. *Computer Graphics*, 21:25-34, 1987
- Winkler, T. Composing Interactive Music. *Cambridge, Mass: MIT Press*, 2001

Natural Language Generation from Concept Blends

Francisco C. Pereira*

* Creative Systems Group AILab - CISUC
Departamento de Engenharia Informática da Universidade de Coimbra
Pinhal de Marrocos 3030 Coimbra
camara@dei.uc.pt

Pablo Gervás†

†Facultad de Informática
Universidad Complutense de Madrid
28040 Madrid
pgervas@sip.ucm.es

Abstract

Conceptual Blending (CB) is a framework for the integration of knowledge from different sources onto one, known as the *blend*. For example, the concept “edutainment” is the blend of “education” and “entertainment”. This paper presents an approach on producing a natural language *reading* of the blends that are generated by Divago, a system that applies a computational implementation of CB to transform a search space. Represented through a semantic network (the concept map), the blend demands considerable effort in its interpretation. Obtaining its Natural Language description raises several issues and is an interesting challenge for the field of Natural Language Generation. In this paper, we propose a reading based on diagnostic features with respect to a reference concept. We make some reflections on the choice of these references, namely regarding the creative potential associated.

1 Introduction

In this paper, we propose a methodology for producing Natural Language texts from knowledge structures called *blends*, which are generated from the combination of other knowledge structures called *input spaces*. These structures (the blend and the input spaces) are represented as concept maps. A concept map should represent the explanation of one or more main concepts by means of a graph structure having directed arcs connecting pairs of nodes, where each arc represents a relation and each node a concept. For example, we can explain the concept of “eating” by association to other concepts like “digesting”, “absorbing”, “food”, “energy” or “survival”. Or a concept like “bird” by association to “wings”, “feathers”, “beak” and so on. From the point of view of Natural Language Generation (NLG), an immediate contribution from concept maps can be as an explanatory core for the lexicon that allows associative reasoning, like metonymy or metaphor. The non-literality emerging from these and other linguistic devices is, we think, an important brick for creativity in language.

The approach we are currently following is based on the principle that, when communicating a new concept in natural language, it is common to use previously established concepts as a support. For example, for describing the concept of “laptop computer”, we do not have to enumerate all the specific features of computers in general, instead we take them all for granted (by just saying that “it is a computer”) and focus the differentiating features (being portable, light, using a battery, etc.). This idea for reading blends is thus based on finding *salient* (Milosavljevic and Dale, 1996) or *diagnostic* (Costello and Keane, 2000) features and using them to describe the concept.

Conceptual Blending (Fauconnier and Turner, 1998) is a framework for integration of knowledge (e.g. “digesting a book” comes from an integration of “eating” and “reading”) and we will explain it in some detail in section 2.2. It should be clear, though, that we are proposing the *reading* of the *blends* that are generated by our computational model of blending, not blends in general or generated by whatever forms the framework of Fauconnier and Turner may take. For our purposes, the first and most important notion to catch now is that of a *domain*, which is represented primarily by a concept map. Therefore, *reading a concept map* is the necessary first step towards *reading a blend*. The other step of exploration is that of *reading instances*, and we should point out that we have already done some work and experiments at this level¹, which took the form of exploring visual constructions obtained from a blended instance, instead of natural language descriptions.

In the next section, we give more information regarding the blending mechanism, after which we dedicate a section to describing existing work on the subject of comparing and describing objects. The general idea of our NLG system is presented in section 4. As the reader will notice, this is clearly a work under development, as also confirmed in the examples shown. We finish some discussion of important issues to be considered.

2 The Blending Mechanism

A detailed description of our implementation of Conceptual Blending is beyond the scope of this paper, however a

¹See the “house-boat visual blending experience” (Pereira and Cardoso, 2002a)

few basic notions about the knowledge representation formalism and the algorithm for obtaining blends are needed to understand the general idea of the paper.

2.1 Concept Maps

Before entering this section, we should clarify that, from the point of view of AI, a concept map is “just” a semantic network, a rather classical knowledge representation structure, in general terms. Yet, we would like to provide the background and philosophy that lead us towards this choice as well as some specific methodology we are applying.

Semantic networks have been used for generating discourse from early on (Simmons and Slocum, 1972; Shapiro, 1982). Novak and Gowin (1984) were the first to focus concept maps under a study on education and communication in humans and they considered them a very intuitive and commonly used graphical representation of knowledge structure in domains. A concept map is a directed graph in which arcs connect pairs of nodes, each node corresponding to a concept, each arc to a relation. This definition covers a very big set of formalisms and applications we can find in AI, namely semantic networks, ontologies – e.g. WordNet (Miller, 1995) –, Semantic Web (Berners-Lee and Miller, 2002), among others. In our current focus, the Divago project (Pereira, 1998), we use concept maps to represent a *domain theory*, the explanation of the concepts of a domain. In principle, our notion of domain is not restricted to any pre-defined abstraction, but to the *user’s* choice, being any concept map she builds around a theme or goal. Yet, for normalization purposes, we are using an extra constraint: all the relations must come from the hierarchies of the Generalized Upper Model (Bateman et al., 1995), a general top-level ontology that has two hierarchies (elements and relations) that comprise abstract relations, properties, spatial relationships, among others. Figures 1 and 2 show an extract of the concept maps of the domains of horse and bird. Each relation is described by a predicate `rel(D, C1, R, C2)`, where the relation *R* is a labelled arc from the concept *C1* to the concept *C2* in domain *D*.

In comparison to Sowa’s Conceptual Graphs (Sowa, 2000), it is clear that the language we use is much more limited: we only allow the valence of 2 arguments (every relation is binary), there is no possibility for specifying a quantifier for each relation, context or descriptor, to name a few differences. This makes clear that making the natural language reading of Conceptual Graphs (CGs) is more straightforward (if the full potential of these is used) than in our concept maps, given the ambiguity that may arise in these.

It is important to say, though, that our specific choice for the concept map representation as opposed to CGs is much more attached to its simplicity (with regard to the blending process) than to any other aspect. Yet, we are currently studying and implementing further evolutions

```
rel(horse, horse, existence, wilderness).
rel(horse, farm, isa, human_setting).
rel(horse, house, isa, human_setting).
rel(horse, snout, pw, horse).
rel(horse, mane, pw, horse).
rel(horse, tail, pw, horse).
rel(horse, leg, pw, horse).
rel(horse, leg, purpose, stand).
rel(horse, hoof, pw, leg).
rel(horse, horse, taxonomicq, ruminant).
rel(horse, horse, eat, grass).
rel(horse, horse, ability, run).
rel(horse, horse, carrier, human).
rel(horse, human, ride, horse).
rel(horse, horse, purpose, cargo).
rel(horse, horse, purpose, traction).
rel(horse, horse, purpose, food).
rel(horse, mane, color, dark).
rel(horse, mane, size, long).
rel(horse, mane, material, hair).
rel(horse, hoof, quantity, 4).
rel(horse, leg, quantity, 4).
rel(horse, eye, pw, snout).
rel(horse, ear, pw, snout).
rel(horse, mouth, pw, snout).
rel(horse, eye, quantity, 2).
rel(horse, ear, quantity, 2).
rel(horse, eye, purpose, see).
rel(horse, ear, purpose, hear).
rel(horse, mouth, purpose, eat).
rel(horse, horse, sound, neigh).
rel(horse, horse, motionprocess, walk).
```

Figure 1: Extract of facts from the *horse* domain

to our representation, namely by allowing quantifiers and context specifiers.

2.2 Conceptual Blending

In the simplest version of Conceptual Blending, as described by Fauconnier and Turner (1998), four distinct *mental spaces* are considered: the *generic space*, which should contain the generic background knowledge of a cognitive agent; two *input spaces*, which should comprise specific knowledge (e.g. the concepts that define horse and bird) that is to be integrated in the blend; and the *blend*, which contains a (potentially) novel concept (or set of concepts), and has a structure that results from the other three spaces and from its own emergent logic. For example, the fictional character of Count Dracula can be analysed as resulting from the blend of “vampire bat” and “evil person”², thus having properties and structure from both input spaces. Yet, while giving *life* to the character, some new features emerge, like “hates garlic” or “only dies if his heart is pierced with a wooden stake”. This process they call “running the blend”. The process of generating the blend demands a mapping correspondence between elements from the input spaces, which, according to a set of principles (the optimality constraints) are subject to be transferred (projected), potentially bringing

²We do not argue this is how Dracula was imagined, yet it serves as an example of the blending process.


```

rel(bird, bird, smaller_than, human).
rel(bird, lung, pw, bird).
rel(bird, lung, purpose, breathe).
rel(bird, paradise_bird, isa, bird).
rel(bird, bird, ability, fly).
rel(bird, feathers, pw, bird).
rel(bird, beak, pw, bird).
rel(bird, beak, purpose, eat).
rel(bird, bird, sound, chirp).
rel(bird, eye, pw, bird).
rel(bird, bird, existence, house).
rel(bird, bird, existence, wilderness).
rel(bird, eye, quantity, 2).
rel(bird, wing, quantity, 2).
rel(bird, claw, quantity, 2).
rel(bird, wing, pw, bird).
rel(bird, wing, conditional, fly).
rel(bird, wing, purpose, fly).
rel(bird, claw, purpose, catch).
rel(bird, leg, pw, bird).
rel(bird, leg, purpose, stand).
rel(bird, claw, pw, leg).
rel(bird, leg, quantity, 2).
rel(bird, parrot, isa, bird).
rel(bird, parrot, ability, speak).
rel(bird, bird, role_playing, freedom).
rel(bird, straw, pw, nest).
rel(bird, oviparous, lay, egg).
rel(bird, bird, purpose, pet).
rel(bird, bird, purpose, food).
rel(bird, eye, purpose, see).
rel(bird, beak, purpose, chirp).
rel(bird, bird, motion_process, fly).

```

Figure 2: Extract of facts from the *bird* domain

attached their surrounding structure. This projection is selective in the sense that some elements are projected to an *identical* existence in the blend, some are projected to a *different* existence (normally their counterpart in the mapping) and some are not projected at all. In the Dracula example, the “big canine teeth” are projected to “big canine teeth”, the “bat wings” are projected to the “man’s cloak”, and the “sonar” is not projected at all.

In our computational model of CB, currently under development, we apply our implementation of those processes and principles (Pereira and Cardoso, 2001, 2002a,b, 2003a) to generate the blend. Each of the spaces is called a domain, and is represented via a concept map, rules, frames, integrity constraints and a set of instances. For the present purposes, the reader need only imagine each domain as being a concept map.

In figure 3, we show a diagram with the four spaces involved in the example we use in this paper: the blend of a “horse” and a “bird”. The reader must understand this is a very simplified diagram (with only three concepts for each domain) in order to make it readable. Figures 1 and 2 have already given an idea of the input spaces, while figure 4 presents a (very small) extract of the generic space, which is expected to contain domain-independent knowledge such as *isa* taxonomies, integrity constraints³ and

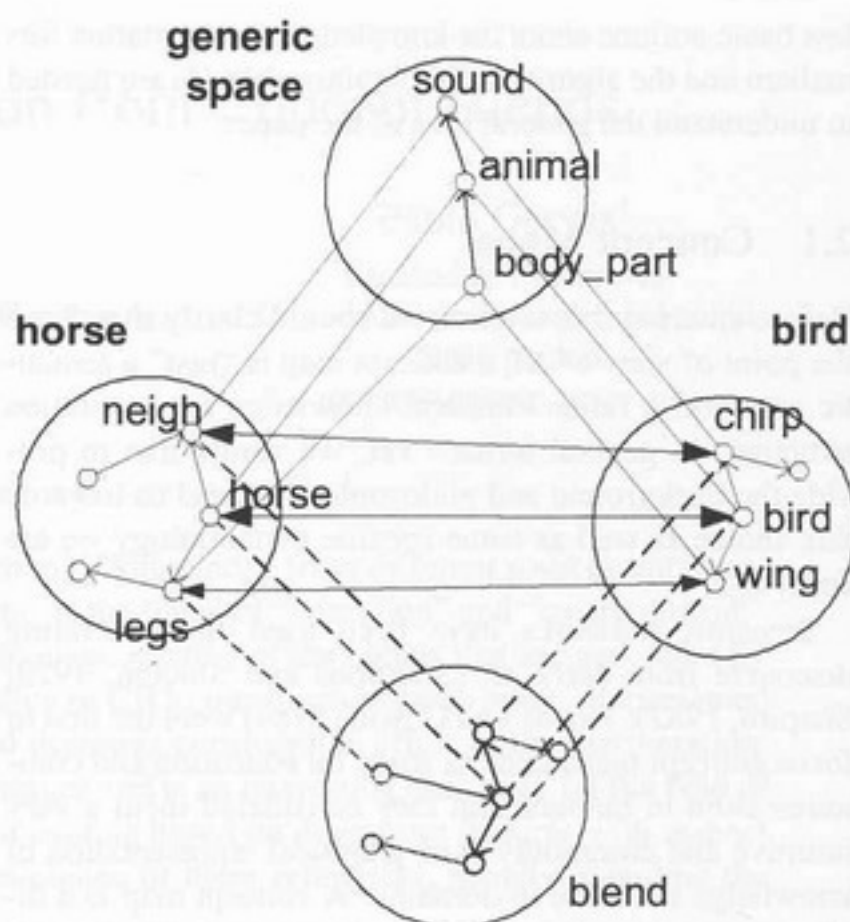


Figure 3: Example of a Conceptual Blending of horse and bird

frames⁴.

It should be obvious that, given the mappings, there is potentially a multitude of projection combinations in the blend (e.g. “a horse that chirps and has wings”, “a bird that neighs and has legs”, “a horse that neighs and has legs”, etc.). This explosion is controlled by the constraints (Pereira and Cardoso, 2003a), yet the search space is very large and complex so we apply a *Genetic Algorithm* to assure a parallel, non-deterministic search mechanism. Thus, we obtain a variety (though considerably reduced when compared to the unconstrained version) of different results (Pereira and Cardoso, 2003b).

In order to get an objective measure on the novel associations originated in the blend, the system finds the two lists of distinctive facts, one for each input domain. Each one of these lists, known as *extra* lists, contains basically all the relations in the blend that do not exist in the domain. Conversely, and also to assess the difference between the blend and the input domains, there are also *missing* lists (also one for each input), which contain the relations that are in the domain, but do not exist in the blend. As the reader will see, these lists are also fundamental for the NLG readings we are discussing here.

In figure 5, we can see an example of a generated blend (in this case, identified as being the individual 10210 of the genetic algorithm run) of “horse” and “bird” and, in figure 6, we show the respective extra and missing lists with respect to the “horse” domain. The syntax of these lists is straightforward: each element has the form $X/R/Y$, where R is the relation and X and Y are the pa-

³The example shows the Divago notation for two: “X cannot have two different shapes at the same time”, and “X cannot exist in two different quantities”

⁴One is shown in the example, which can be interpreted as: “In the Blend, Concept X has an ability A, which didn’t exist before in X’s original domain, D1. It is P, a subpart of X, which enables this ability”.

```

rel(generic, entity, isa, something).
rel(generic, situation, isa, something).
rel(generic, temporal_entity, isa, entity).
rel(generic, spatial_entity, isa, entity).

:

rel(generic, person, isa, human).
rel(generic, human, isa, animal).
rel(generic, animal, isa, living_entity).
rel(generic, speed, isa, property).
rel(generic, size, isa, property).
rel(generic, color, isa, property).
rel(generic, weight, isa, property).
rel(generic, behavior, isa, property).
rel(generic, heavy, isa, weight).

:

integrity(generic, [shape(X, Y),
                    shape(X, Z),
                    {Y\=Z}], []).

integrity(generic, [quantity(X, Y),
                    quantity(X, Z),
                    {Y\=Z}], []).

:

frame(generic, new_ability(D1),
      [ability(X, A), purpose(P, A), pw(P, X),
       {current_blend(Blend),
        projection(Blend, D1, X, X),
        other_input_domain(D1, D2),
        projection(Blend, D2, A, A)},
       op(exists([projection/D1/X/X,
                   projection/D2/A/A])),
       [{rel(D1, X, ability, A)}],
       [new_ability(X, A)], []).

```

Figure 4: The generic domain

rameters – e.g. the element `wing/conditional/fly` in the extra list means the “wing” in the blend has a “conditional” relation with `fly`, a relation that did not exist in the “horse” input domain, i.e. there is no `rel(horse, wing, conditional, fly)`.

We are thus faced with a fundamental problem: how to “read” the results? In a previous work, we have already approached the visual reinterpretation of the instances according to the new associations. We applied CB to the domains of “houses” and “boats”, and re-generated the drawings we had for instances of those domains (e.g. “A boat has a window”, being the representation of a “window” given by a square, implied the drawing of the “boat” would have a “hatch” of square form instead of round).

Now, we are interested in the understanding of the concept maps themselves via natural language descriptions.

```

rel(ind10210, tail, pw, horse)
rel(ind10210, leg, pw, horse)
rel(ind10210, leg, purpose, stand)
rel(ind10210, hoof, pw, leg)
rel(ind10210, horse, taxonomicq, ruminant)
rel(ind10210, horse, eat, grass)
rel(ind10210, horse, ability, run)
rel(ind10210, horse, carrier, human)
rel(ind10210, human, ride, horse)
rel(ind10210, horse, purpose, cargo)
rel(ind10210, horse, purpose, traction)
rel(ind10210, horse, purpose, food)
rel(ind10210, mane, color, dark)
rel(ind10210, mane, size, long)
rel(ind10210, mane, material, hair)
rel(ind10210, hoof, quantity, 4)
rel(ind10210, leg, quantity, 4)
rel(ind10210, eye, pw, snout)
rel(ind10210, ear, pw, snout)
rel(ind10210, mouth, pw, snout)
rel(ind10210, eye, quantity, 2)
rel(ind10210, ear, quantity, 2)
rel(ind10210, eye, purpose, see)
rel(ind10210, ear, purpose, hear)
rel(ind10210, mouth, purpose, eat)
rel(ind10210, horse, sound, neigh)
rel(ind10210, horse, motion.process, walk)
rel(ind10210, horse, ability, fly)
rel(ind10210, wing, purpose, fly)
rel(ind10210, wing, pw, horse)

```

Figure 5: Blend generated for *horse* and *bird*

```

Extra:
[wing/conditional/fly, wing/pw/horse,
horse/motion.process/fly, horse/ability/fly,
wing/purpose/fly]
Missing:
[]

```

Figure 6: Extra and missing lists for the *horse* domain

3 Generating Descriptions and Comparisons

Two lines of existing research had to be taken into account before attempting the task outlined above: work on establishing comparisons between objects at the conceptual level, and work on generating natural language descriptions of objects from their conceptual definition. In some cases, actual research work had brought together both endeavours.

3.1 Establishing Comparisons between Objects

In their C3 model of Conceptual Combination (Costello and Keane, 2000), the authors present three constraints (*plausibility*, *diagnosticity* and *informativeness*) for the interpretation of noun-noun compounds. It is not within the scope of this paper to explain or compare them to our approach, yet we would like to retain diagnosticity for the present analysis. This constraint consists on identifying the differentiation (or diagnostic) aspects of a concept

in comparison to its neighbouring concepts. For example, the diagnostic features of a cell phone are that it is a phone that is (normally) very small, wireless and one can speak with it if we are within the range of the provider's "cell". In this case, non-diagnostic features could be its colour, having a keyboard or being a communication device, which could at best say it is a phone. The idea of diagnosticity has much in common to the work of Milosavljevic (1997a,b) around salience properties of concepts. An interesting observation on salience is that it is dependent on the user-model (geographic location, level of expertise in the domain, etc.), thus differentiating properties depend on the relations users have with the concept (e.g. describing a "llama" to two kids by comparison to the animals they already know may bring salience to different sets of properties). We think the philosophy behind salience and diagnosticity is the same, yet its realization is different. In Milosavljevic (1997a,b), the authors hard-code salience ranking (some properties should be more salient than others, according to user models), as well as many other features of the system, which was justifiable given the specific application they had in mind, a dynamic hypertext system (Peba-II) for description of animals. In Costello and Keane (2000), the domain is that of nouns in general, and so finding diagnostic properties was done by automatic mining the corpus of all nouns in memory, not giving special value to specific properties or relations. Still, we think, if the nouns were animals that a kid knew about (as in the example just given), the results would be similar to those of Peba-II, and therefore diagnosticity would be the same as salience. An important feature in Milosavljevic's work, which is directly related to our present concern, is that Peba-II can describe an unknown entity, the *focused entity* (e.g. an echidna), by comparison to a known entity, the *potential confusor* (e.g. a porcupine) generating a text that enhances the *common salience properties* and the *contrasting salience properties*.

3.2 Generation of Descriptions in Natural Language

Research on natural language generation over the years has come to propose a pipelined architecture (Reiter, 1994) as the simplest engineering solution to generate texts meant to convey information. This solution is not optimal and the generation of other types of texts calls for different architectures (DeSmedt et al., 1995; Beale et al., 1998). However, for the present purposes, the modularity presented by a pipeline architecture outweighs any other disadvantages that it may have.

The pipelined architecture establishes a number of basic tasks to be carried out when generating a natural language text: *content determination* - finding what to say -, *document structuring* - organising what is to be said -, *sentence aggregation* - grouping together the parts that allow it -, *lexicalization* - selecting the words that will realize each concept -, *referring expresion generation* -

choosing the right expression to refer to each element in its actual context - and *surface realization* - turning the result into a linear natural language sentence. These tasks tend to be grouped into bigger modules that operate in sequence over the initial input: a *text planning* component that deals with content determination and document structuring, a *sentence planning* component that deals with aggregation, referring expression generation and lexicalization, and a *surface realization* component. In this paper we focus particularly on text planning, in the sense that the innovative contribution lies in the way conceptual blending is applied to solve part of this task. The rest of the tasks are solved by resorting to state of the art solutions, in most cases by simple expeditive methods borrowed from existing work.

Natural language techniques have been applied in the past to generate textual descriptions of data that existed in a digital form that was not easily readable by humans, for domains like describing object-oriented models (Lavoie et al., 1997), information stored in a museum database (Dale et al., 1998), or time series data such as meteorological reports (Sripada et al., 2001).

Of particular interest is the body of work on natural language generation from conceptual graphs (Velardi et al., 1988; van Rijn, 1992; Smith et al., 1994; Svenberg, 1994; Wagner et al., 1995; Nicolov et al., 1996). These efforts take advantage of the well defined operations of conceptual graphs to explore the task of generation. Although some of the insights obtained in this way are applied in the present paper, the differences in complexity between this knowledge representation formalism and the concept maps used here required that further exploration of possible connections be postponed until a basic generation capability was integrated with Divago's process of conceptual blending.

For the current endeavor, an existing surface realization module, FUF/SURGE, (Elhadad, 1993) has been used to take care of the final stage. This delegates the task of dealing with the linguistic knowledge required to linearize and realize the text plans resulting from the previous stages into their final form as natural language sentences. The work presented here deals therefore with the task of planning an NL realization for a given concept map.

The communications between the different modules involved takes place in terms of the generic data structures defined for generic architectures for NLG systems in the context of the RAGS project (The Rags Project, 2001).

4 Reading Concept Blends

In our first approach, we generated a reading by transcribing individually each relation of the resulting concept map. The resulting factual description of a domain, a list of sentences, with one sentence for each fact, made arid reading. We thus arrived to the conclusion that, unless

with intensive knowledge and processing, this method was bound to arrive at very unnatural outcomes. The dead-end we apparently had encountered lead us to reversing the direction, i.e., instead of starting with the maximum possible of sentences (one for each relation/fact), we decided to look for minimal descriptions, and eventually elaborate them towards a “natural” description.

Usually, humans refine these descriptions by grouping together sentences about related facts, constructing chains of discourse that can be read as a related paragraph. This task is simplified by taking into account additional information such as a *discourse goal* - what goal is to be addressed by the generated text -, a *user model* - what are the particular needs of the user for which the text is intended -, or a *discourse history* - which information has already been conveyed by the system to the user. This information can be used to deterministically plan the required text. Unfortunately, in attempting to provide a reading for a concept map in no specific context, such information is available only in a sketchy form. At best we can say that the discourse goal, taken here as default, is to describe the domain expressed in the concept map in a way that makes it easy to apprehend for a human reader. We are adding a further constraint to the extent that we are trying to obtain the minimal description that will identify to the user the relevant differences between the focused entity and the potential confusor. The conceptual map that describes the domain that has been chosen as referent (Milosavljevic’s potential confusor) is in fact acting as a reference for what the user already knows. In this sense, it can be understood as encoding the information about the user that is relevant for the task.

4.1 Text Planning

The text planning component must provide a plan for the text that is to be generated. In our case, this plan must be a semantic specification of what is to be said to describe the concept blend that has resulted from a process of Conceptual Blending as described above.

The rationale we are trying to bring here is that a “natural” procedure that humans regularly apply for describing concepts is by recurring to the diagnostic (or salient) features, and this leads us to the stated goal of “reading a blend”. Once we identify a reference concept (or Milosavljevic’s potential confusor) that is close enough to the blend, we are more likely to build a “natural” reading. This reading can be creative in as much as the system is able to find unexpected references. For example, a blend of a horse and a bird can be read as “a horse that has a pair of wings to fly” or as “a plane without an engine and that uses food instead of fuel” (both can carry humans and have a pair of wings and fly), the former taking a horse as reference and the latter taking a plane. Of course, these more “creative” readings demand a space of choices with higher complexity (in the plane example, some features are ignored, some are emphasized). Here, taking a less

creative starting point, we propose the use of the input spaces as references for the description, and this will consist on the use of the extra and missing lists as a source for diagnostic features.

Apart from exploring the referred creative aspects, the practical goal of the application of diagnosticity in the NLG readings of the blends is twofold: to provide an intuitive reading of the blend, which, if made by hand in a relatively small concept map, demands considerable effort; and to help in the validation of the system, when confronted with Costello’s empirical experiments with people, in which the description follow the same kind of reasoning (e.g. “A X Y is a Y with this and that feature of X, without this and that feature of Y...”).

In the ideal case, the text planning module would receive as input: the concept maps that describe the domains that have been blended, the blend, and the statistics generated by Divago while running the blend. As explained above, we intend to describe the blend by comparison with one of the original domains. One of the two possibilities must be chosen. We have opted for taking the domain that is most similar to the blend. Alternative options (like focusing on the most different domain) are possible, and they will be explored in detail in further work. However, at the present level of refinement of the NLG modules, they resulted in output that was too verbose to be processed fruitfully, in the sense that it entailed re-describing a high proportion of the most similar concept.

During the blending process Divago carries out various operations that entail comparing the input domains and searching for analogies between them. The results of these operations are present in its output in the form of additional statistics regarding the observed characteristics. These statistics are generated with respect to both input domains. Once we have chosen the most similar one, we concentrate on the statistics that describe the differences between it and the blend: the list of facts in the blend that were not in the original concept map (the extra list), and the list of facts that were in the original concept map and have disappeared from the blend.

Additionally, in order to produce adequate natural language output, the system must be provided with a name for the concept that has been chosen as reference, and a name for the blended concept. These are used by the system to identify data from each domain when generating the description.

The system operates by structuring the output in terms of a simple schema for comparison. First the comparison is established between the blend and the chosen referent (i.e. “A horsebird is a horse”), then the description is generated by processing the extra and missing lists.

The process of converting those lists into semantic descriptions of the intended content follows a number of steps that progressively refine the input.

The first step involves transforming the information conveyed by the facts in the extra and missing lists into

blocks of information susceptible of being converted at a later stage into linguistic expressions. The representational simplicity of the concept maps used by Divago is oriented toward easing the task of conceptual blending, but it is at times ill-suited for linguistic expression. For instance, a concept map may contain separate facts to represent the following information about a bird: that a bird has wings, that the number of its wings is two, that the purpose of its wings is flying. In fact, the information linking the number or the purpose of wings is only implicitly linked to the bird by the co-occurrence of the corresponding facts in the description of the same domain. At this stage of the processing, such implicit information (that a bird has two wings for flying) is made explicit, and expressed in terms that the rest of the modules of the system will later recognise. This operation represents an elementary kind of aggregation at the conceptual level. The output of this process is already divided into units that will later correspond to linguistic clauses. Because the description is to be phased in terms of a comparison between the two main concepts involved (i.e. a "horse" and a "horsebird"), this process of aggregation is geared towards generating clauses in which the main concept acts as the head of the clause. This operation is akin to the *incremental construction* used for generating sentences from conceptual graphs in Nicolov et al. (1996), in the sense that the facts corresponding to the original concept map are progressively consumed to produce semantic representations that already correspond to sentences. The order of consumption is determined by priorities over the mapping rules, to ensure that those rules resulting in maximum aggregation are applied first.

Once all the facts in each of the lists have been condensed into semantic formulations as concise as possible, a further step of aggregation is applied at the semantic level. For each list, the semantic descriptions are grouped together whenever two of these clauses share a subject or an object, forming a single clause with conjunctive expressions as subjects or objects.

Having obtained the most compact version possible of these lists, a single list is built by appending all the facts in the extra list and the list formed with the negations of all the facts in the missing list.

The resulting list is then submitted to a further process of semantic aggregation to ensure the most concise solution possible.

The output of the text planning module is converted into a simple rhetorical representation where the resulting list of semantic representations of clauses is included as elaborations on the opening statement ("A horsebird is a horse").

4.2 Sentence Planning and Surface Realization

Little need be said about the sentence planning and surface realization components. Basic state-of-the art pro-

cessing is applied to eliminate redundant mentions of the same concept during reference, substituting by referring pronouns wherever possible. The size and complexity of the text plans that are being obtained with the current inputs do not require further complication. This may change in the future as more elaborate cases are considered.

The lexicalization module enriches the terms that have been employed so far to include all the additional information that is required by the FUF/SURGE surface realization module. The list of sentence representations received by this module is converted into Functional Descriptions of FUF. Although the conversion that takes place in this module is straightforward look-up on a one-to-one semantic-lexical association table, this does not necessarily imply that there is no *true lexical choice* (Cahill, 1999) involved. For each of the lexical realizations that the system provides in its output, there are usually alternatives that it might have chosen. The fact that choices are limited at this stage is due to the strict criteria applied during content determination, where semantic constructions that might lead to lexicalizations in which the main concepts are not heads of the respective clauses are pruned from the text plan in favour of others. In this sense, the decisions that govern lexical choice are taken in previous stages, and the actual lexicalization of the each sentence plan is reduced to *lexical realization* (Cahill, 1999).

The output of the final module is a sequence of natural language sentences as shown in the examples.

4.3 Results

The procedure described here was applied to thirty conceptual blends generated by Divago from the concept maps for a "horse" and a "bird". In each case, the output was a concept map describing the blend and the set of statistics that describe the relationships between the resulting blend and each one of the original domains.

Not all the resulting blends correspond to concepts that are interesting to describe from our point of view. For some reason, Divago seems to prefer to project facts from the "bird" domain onto the "horse" domain, and the resulting blends are therefore all to be described by comparison with a horse. No single blend resulted where the most similar concept was a bird. This is related to the way in which the integrity constraints and the frames in the generic space interact when running the blend. In some cases the only facts that are projected from the bird domain onto the blend are impossible to relate with the facts about horses in the original domain (e.g. information about nests appears is projected the blend, by no explicit relation between "horsebirds" and nests appears). In such cases, the automatically generated descriptions come out as nonsensical.

In other cases, mismatches between the level of detail of the descriptions between the two domains result in un-

- (1) A horsebird is a horse. A horsebird can fly, it has feathers, a beak, and wings for flying and it moves by flying.
- (2) A horsebird is a horse. A horsebird has four eyes, it can fly and it has wings for flying and moves by flying. A horsebird does not have four legs.
- (3) A horsebird is a horse. A horsebird has hooves for flying, it can fly and it moves by flying.
- (4) A horsebird is a horse. A horsebird has two wings and feathers. It can fly, and it moves by flying.
- (5) A horsebird is a horse. A horsebird can fly. It chirps, it has wings for flying and it moves by flying.
- (6) A horsebird is a horse. A horsebird has wings for flying, it can fly, it is smaller than a human and it moves by flying.

Figure 7: Example readings for the *horse bird* blends

interesting projections (e.g. the “horse” domain does not mention that horses have lungs, whereas this information is included for birds; Divago considers attributing lungs to a horse as an interesting blend, since, according to the information in the “bird” domain and the frame in the generic space, this will give a horse the ability to breathe).

The selection of readings presented in figure 7 constitutes a selected sample of the total, in which those readings which were uninteresting for reasons such as the above have been eliminated.

Although these readings are nothing out of the ordinary from a linguistic point of view, they have already proved very useful as simple tools for providing a better understanding of the peculiarities of the blends that Divago is currently producing. The observations about the tendency towards one rather than the other domain, or the differences in level of detail between domains already provide insights that will allow further trimming and refinement of the algorithms.

From a more general point of view, better results will be obtained after several iterations of such refinements. The reason for the need of several iterations is that, in truth, what we have is a natural language generation process in which the text planning process is distributed between the actual operation of blending carried out by Divago and the text planning module. The first step of iteration might simply revise the processes of transcription onto semantic units, of aggregation at the various levels, or of lexicalization. However, better results may be obtained if the refinements are taken a step further, to affect the actual representations employed for each domain (the concept maps) or the set of integrity constraints and frames that are provided in the generic domain.

5 Discussion and Further Work

Contrasting the *blend* with the input spaces is possibly the clearest way to understand its meaning, and thus provide

a natural language reading. This observation could be extended to every unknown concept described via a representation similar or equivalent to a concept map. This leads us to the description of concepts by comparison, a field already being explored by others. We are not sure whether the work presented here provides any specific contribution to this field, other than being another practical example of its application. Yet, we would like to bring a short discussion around the creative aspects of the description of concepts by comparison.

The first aspect is related to the choice of the reference. Above, we gave the example of the two choices of reference for describing a “horse-bird” (as the one showed in figure 5): the input domain with shortest extra and missing lists (the “horse”) and an apparently “distant” one, the “plane”. Although without having implemented the “plane” data, we believe using the “bird” domain as a reference to that example of figure 5 seems to us a less natural or creative description. Namely because a plane and our “horse-bird” both fly, have wings, carry and are driven by humans, they end up being semantically closer than initially expected. The interesting fact now is that the *reading* with the plane as a reference still seems unnatural or, at the least, a non-literal interpretation of the concept map. In other words, and this is a (bold) conjecture demanding further study, choosing as reference the closest input space (the one with smaller missing and extra lists) provides the clearer insight on the blend, while the choice of a third entity (other than the input spaces) may bring alternative readings, enhancing other aspects or triggering metaphoric associations. Proving this hypothesis will demand the application of the reading mechanism described here to a wide range of blends, input spaces and “external” references.

Another aspect that may reveal creativity in such a system lies in the similarity abstraction between the blend and the reference. In the present stage, this similarity is based on *pure* set difference, i.e., we compare the sets of relations from the blend and from the reference. Yet, other levels of similarity, such as those based on structure may provide additional insights. For instance, in the “trashcan basketball” example, it is not correct to say that “trashcan basketball is a basketball game where...”, yet these two “games” have some structural similarities that must be explored in the reading (e.g. there is something being shot towards a container; the goal in both is to get as much shots in the container as possible, etc.). Here we see *structure similarity* as happening when we can find a mapping between objects between the domains based on identical relational structure (e.g. having “person shoots crumpled paper” and “player shoots ball”, we have a mapping between “crumpled paper” and “ball” and “person” and “player”). The textual interpretation of these cases seems more complex than in set difference comparison and, as in the discussion of reference choice, there may be a multitude of options to choose from. This discussion leads naturally to other less central intentions we have be-

hind this work, which we will explore in further developments: to move from the static reference reading – “X is a Y with...”, Y being the reference – to metaphoric reference where a concept is described with another’s structure – e.g. “In trashcan basketball, the player shoots the crumpled paper to the waste basket in order to score”⁵ –, the latter demanding a more subtle reasoning. On this issue we intend to explore possible synergies with the work of Jones (1994) regarding the generation of transparently-motivated metaphors.

From the point of view of natural language generation, the present work was faced with the obstacle of not having a set of human-generated natural language descriptions of the domains that were to be described. Where such a source is available, knowledge acquisition techniques can be applied either to the corpus of examples or to the experts that produced them, and a wealth of information about how the descriptions are constructed by humans can be obtained in this way (Sripada et al., 2001). The kind of information that may be obtained in this way includes text-level resources such as discourse-plans (Dale et al., 1998) which help to provide a pre-designed structure along which to organise the content. In the present case, no such corpus was available, so the text planning process is currently a freely creative process, with ongoing work progressively elaborating useful heuristics to guide it along the vast search space of possible organizations of the descriptions that can be extracted from the facts.

The question of how to evaluate the performance of natural language generation applications is still an open problem. For a related text planning problem, Sripada et al. (2001) propose an evaluation set up where the results are evaluated at the content level as well as at the text level. This involves having a conceptual representation of the texts generated, and having humans construct equivalent representations based on the same input data. Comparison of human and machine generated conceptual representations (as well as human evaluation of the output texts) would allow to calibrate the system both in terms of the quality of text generated and the appropriateness of the organization chosen for the content. As we said earlier, we plan to use Costello and Keane’s examples, and compare their textual descriptions (e.g. “a parrot robin is a parrot that can talk”; “a rhinoceros horse is a horse with horns”) with our readings. This will provide us with an estimate of how “natural” our automatically generated readings are.

From the NLG point of view, when covering also “external” domains (such as the “plane” and many others), we will be able to produce textual descriptions and finally study our conjectures around the creative aspects of this work.

In the Conceptual Blending system, apart from keeping the current developments on the implementations, we have now identified some fundamental work to do,

namely towards the extension of the concept maps and using already built knowledge bases (like WordNet) to make wider range tests. Another interesting idea to follow is that of using the frames as references for NLG description. In Divago, a frame is a structure that allows the definition of top-level concepts or patterns. For example, a frame can specify the conditions a concept must satisfy in order to be a “transport means”. This leads to the use of several references (from frames, external domains, input spaces) simultaneously to participate in the natural language generation of a blend description.

Acknowledgements

The authors are grateful to the anonymous reviewers for their helpful comments, which have greatly improved the original draft.

This work was partially supported by grant TIC2002-01961 of the Spanish Comisión Interministerial de Ciencia y Tecnología, and COST initiative 282 of the European Union.

References

- J. Bateman, B. Magnini, and G. Fabris. The generalized upper model knowledge base: Organization and use. *Towards very large knowledge bases: knowledge building and knowledge sharing*, 1995.
- S. Beale, S. Niremburg, E. Viegas, and L. Wanner. De-constraining text generation. In *Proceedings of the Ninth International workshop on Natural Language Generation, Niagara-on-the-Lake, Ontario*, 1998.
- T. Berners-Lee and E. Miller. The semantic web lifts off. *ERCIM News*, (51), 2002.
- L. Cahill. Lexicalization in applied NLG systems. Technical Report ITRI-99-04, Information Technology Research Institute, University of Brighton, 1999.
- F. J. Costello and M. T. Keane. Efficient creativity: Constraint-guided conceptual combination. *Cognitive Science*, 24(2):299–349, 2000.
- S. Coulson. *Semantic Leaps: Frame-shifting and Conceptual Blending in Meaning Construction*. New York and Cambridge: Cambridge University Press, 2000.
- R. Dale, S. J. Green, M. Milosavljevic, C. Paris, C. Verspoor, and S. Williams. The realities of generating natural language from databases. In *Proceedings of the 11th Australian Joint Conference on Artificial Intelligence*, Brisbane, Australia, 1998.
- K. DeSmedt, H. Horacek, and M. Zock. Architectures for natural language generation: Problems and perspectives. In G. Adorni y M. Zock, editor, *Trends in natu-*

⁵ See Coulson (2000) for more on the “trashcan basketball” example.

- ral language generation: An artificial intelligence perspective, pages 17–46. Berlin: Springer, 1995.
- M. Elhadad. *Using argumentation to control lexical choice: a functional unification-based approach*. PhD thesis, University of Columbia, 1993.
- G. Fauconnier and M. Turner. Conceptual integration networks. *Cognitive Science*, 22(2):133–187, 1998.
- M. A. Jones. *Transparently-motivated metaphor generation*. PhD thesis, University of Delaware, 1994.
- B. Lavoie, O. Rambow, and E. Reiter. Customizable descriptions of object-oriented models. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, pages 253–256, Washington, DC, 1997.
- G. A. Miller. Wordnet: A lexical database for English. *Communications of the ACM*, 38(11), 1995.
- M. Milosavljevic. Augmenting the user's knowledge via comparison. In *Proceedings of the 6th International Conference on User Modelling*, Sardinia, Italy., 1997a.
- M. Milosavljevic. Content selection in comparison generation. In W. Hoeppner, editor, *6th European Workshop on Natural Language Generation (6th EWNLG)*, pages 72–81, 1997b.
- M. Milosavljevic and R. Dale. Strategies for comparison in encyclopdia descriptions. In *Proceedings of the 8th. International Workshop on Natural Language Generation (INLG '96)*, pages 161–170, Herstmonceux, England, 1996.
- N. Nicolov, C. Mellish, and G. Ritchie. Approximate generation from non-hierarchical representations. In *Proceedings of the 8th. International Workshop on Natural Language Generation (INLG '96)*, pages 31–40. Herstmonceux, England, 1996.
- J. D. Novak and D. B. Gowin. *Learning How To Learn*. Cambridge University Press, 1984.
- F. C. Pereira. Modelling divergent production: a multi domain approach. In *European Conference of Artificial Intelligence, ECAI98*. IOSPress, 1998.
- F. C. Pereira and A. Cardoso. Knowledge integration with conceptual blending. In *Proceedings of AICS 2001*, 2001.
- F. C. Pereira and A. Cardoso. The boat-house visual blending experience. In *Proceedings of the Symposium for Creativity in Arts and Science of AISB 2002*. AISB, 2002a.
- F. C. Pereira and A. Cardoso. Conceptual blending and the quest for the holy creative process. In *Proceedings of the Symposium for Creativity in Arts and Science of AISB 2002*. AISB, 2002b.
- F. C. Pereira and A. Cardoso. The horse-bird creature generation experiment. In *submitted*, 2003a.
- F. C. Pereira and A. Cardoso. Optimality principles for conceptual blending: A first computational approach. In *Proceedings of the Symposium for Creativity in Arts and Science of AISB 2003*. AISB, 2003b.
- E. Reiter. Has a consensus NL generation architecture appeared, and is it psychologically plausible? In David McDonald and Marie Meteer, editors, *Proceedings of the 7th. International Workshop on Natural Language generation (INLWG '94)*, pages 163–170, Kennebunkport, Maine, 1994.
- S. Shapiro. Generalised augmented transition networks grammars for generation from semantic networks. *Computational Linguistics*, 2(8):12–25, 1982.
- R. Simmons and J. Slocum. Generating english discourse from semantic networks. *CACM*, 2(8):12–25, 1972.
- M. Smith, R. Garliano, and R. Morgan. Generation in the LOLITA system: An engineering approach. In *Proceedings of the 7th. International Workshop on Natural Language Generation*. 1994.
- J. F. Sowa. *Knowledge Representation: Logical, Philosophical and Computational Foundations*. Brooks Cole Publishing Co., Pacific Grove, CA, 2000.
- S. Sripada, E. Reiter, J. Hunter, and J. Yu. A two-stage model for content determination. In *Proceedings of ENLWG-2001*, pages 3–10, 2001.
- S. Svenberg. Representing conceptual and linguistic knowledge for multi-lingual generation in a technical domain. In *Proceedings of the 7th. International Workshop on Natural Language Generation*. 1994.
- The Rags Project. The RAGS reference manual. Technical Report ITRI-01-07, Information Technology Research Institute, University of Brighton, 2001.
- A. van Rijn. Generating language from conceptual dependency graphs. In T. Nagle, J. Nagle, L. Gerholz, and P. Eklund, editors, *Conceptual Structures: Current Research and Practice*. Ellis Horwood, 1992.
- P. Velardi, M. T. Pazienza, and M. De'Giovannetti. Conceptual graphs for the analysis and generation of sentences. *IBM Journal of Research and Development*, 32(2), 1988.
- J. Wagner, R. Baud, and J. Scherrer. Using conceptual graphs operations for natural language generation in medicine. In *Proceedings of the 3rd International Conference on Conceptual Structures (ICCS '95)*. Springer, Santa Cruz, 1995.

A Connectionist Approach to Driving Chord Progressions Using Tension

Andrés F Melo*

*Department of Computing
City University, London EC1V 0HB
andres@soi.city.ac.uk

Geraint Wiggins

†Department of Computing
City University, London EC1V 0HB
geraint@soi.city.ac.uk

Abstract

This paper describes how a backpropagation neural network learnt and reproduced harmonic patterns. These patterns were related in learning and production to musical tension. The neural network thus learnt to produce the next chord in a sequence such that listeners perceive a given degree of musical tension. During this research, it was confirmed that musical tension is measurable and that harmony conveys tension. Ultimately, the results pragmatically show that harmonic tension patterns exist, can be found and reproduced within the style defined by adequate pieces of music.

1 Introduction

The original intention of this project was as a step to produce music in real time—achieving a balance between the unity and variety in the output. The idea is that a neural network provides the unity by learning the harmonic context from a given piece of music. This harmonic context is further related to a measure of musical tension. By varying the tension level in the input, the output would be expected to stay within the harmonic context of the original (providing unity), and the tension would drive the harmony, thus becoming the source of variety.

1.1 Tension in music

Tension can be seen as the incompleteness of the music at any one point. More intuitively, the degree of tension can be related to how *unfinished* the piece of music would sound if it stopped at the point. Tension thus changes dynamically as the piece of music progresses (Krumhansl, 1997).

Tension has been studied for different aspects of pieces of music. Significantly, Narmour (1990) approaches the effect of melody in tension has been approached analytically. The concept of tension is opposed to that of relaxation, and derives from certain patterns in the melodic lines that produce or solve expectancy.

As for harmony, Krumhansl (1997) measured tension and found that the predictors of chord distance defined by Lerdhal (1988) related to the tension curves well for a piano sonata of Mozart. This suggests that harmony has an effect on tension that is composed with the tension resulting from the melodic lines and presumably from other

aspects of the music.

1.2 Machine learning

Machine learning has been previously used to learn from a set of examples to predict the next chords in a sequence (Bharucha and Todd, 1989; Berger and Gang, 1997). This work extends the concepts by testing whether a backpropagation neural network can learn patterns of tension in harmony and use them to generate sound sequences with new, specific tension curves.

Machine learning requires examples of the behaviour the system is expected to learn. It is in fact ideal when examples are easier to find than rules (Dolson, 1989). Examples of real pieces of music are of course available in various formats, and the choice of a representation will have an effect on the learning curve. The representation used should favour harmony over other aspects of the music used.

Tension curves that run in parallel to the music examples are required as well in this case, as they are part of the behaviour that the network should learn. In this case, the data needs to come from human listeners (rather than from theoretical models), as part of the interest in this work is in letting the network find patterns in the *raw* tension curves rather than reproducing rules that were stated by the models in the first place.

The results of neural networks tend to be pragmatical, and so they tend to be considered to be black boxes. The primary analysis of the results is thus whether the behaviour wanted—the production of a sequence of chords that produces a given tension curve—can be reproduced by the network. However, backpropagation networks allow analysis and rule extraction to some degree. This is achieved by analysing the weights of the links, comparing the performance of different architectures and repre-

*The authors were affiliated to the Department of Artificial Intelligence of the University of Edinburgh when developing the work presented here.

sentations, and by simulation of basic cases. In this case, the information provided by the network will suggest the tension patterns present in the music used in the training example.

1.3 Procedure

This paper concentrates on the results obtained with a musical piece: the fourth movement of Prokofiev's Classical Symphony. This musical piece was selected because the rhythmical and melodic patterns are relatively uniform, and the chords change frequently (often every other bar) and its harmony appears to drive the tension. Its duration (about four minutes) is substantial, but without compromising seriously the attention of the listeners.

This work consisted of three distinct stages:

1. elicitation of tension curves,
2. development and training of the neural network, and
3. evaluation of the output of the neural network.

The following sections address each stage separately. Melo (1998) presents further detail of the first two stages.

2 Stage 1: Eliciting tension curves

To obtain tension curves for the musical piece, ten Western listeners were asked to indicate the tension they perceived while listening to it. They indicated the tension by rotating a sprung wheel (Figure 1). The reaction of the spring provides feedback on the position of the wheel, so the listeners have to apply more force to indicate higher tension, or no force for no tension. Before the Prokofiev piece of music, listeners used the equipment to record the tension in one to four pieces of music (according to random settings of the test), to make sure that they felt comfortable with the equipment and task.

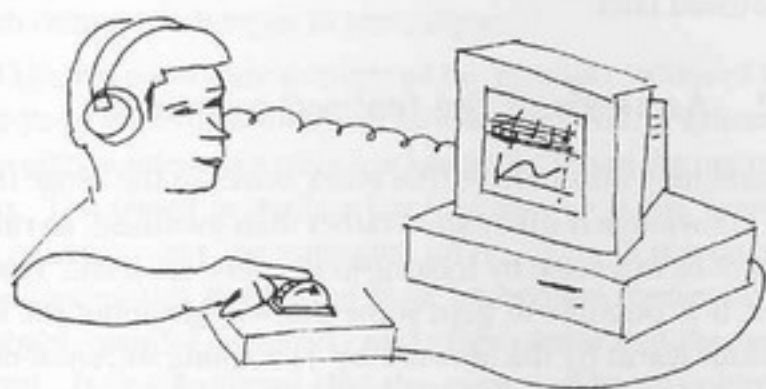


Figure 1: Settings for measuring tension

Before starting the measurement, the concept of tension was explained to the listeners as the uneasy—as opposed to relaxed—sound of the music, and as how “unfinished” the piece would sound if it stopped at that precise

moment. Some of the listeners expressed having initial doubts about their ability to perform the task, but felt confident that they could do it after the first trial.

The music used was a quantised MIDI file in which all the instruments were set to piano and the dynamic was set to be constant. This insures that changes in volume, colour and interpretation are not factors in the perceived tension.

2.1 Raw tension curves

The tension curves given by the listeners were different from each other in that some used the whole range of the wheel whereas others would indicate changes in tension with very slight movements. This was to be expected, as the measure of tension is not defined in absolute terms. Still, the listeners seemed to react at the same points and in the same ways, reacting in the same points, increasing or decreasing the tension at the same time. Visually, the peaks and valleys in the curves appear to coincide (see Figure 2).

The agreement among the curves was confirmed using Friedman's test of rank correlation. Friedman's test compares a number of data series for monotonic behaviour, that is, tests whether the series increase (or decrease) values at the same points (Brownlee, 1965; Hollander and Wolfe, 1973). This test differs from correlation analysis in that it is *non-parametric*: it does not assume that the data series have the same distribution. This allows for the fact that the tension curves provided by different listeners may be expressing their perception of tension in different ways—that is, not necessarily proportionally. Friedman's test showed that the possibility of the curves not being correlated is under 10^{-6} .

2.2 Obtaining a single tension curve

The ten tension curves obtained have to be condensed into one to be used for training. Condensing the data is expected to make up for differences in the “style” of the curves provided by different listeners, as well as for errors in the curve because of a listener losing attention.

The median is used to condense the data series. The argument is the same as when choosing Friedman's test: that the interest is to show that the curves are monotonic. The median is non-parametric because it does not assume a data distribution in the series. To apply it, however, it was found appropriate to normalise the ten curves obtained by scaling them between 0 and 100.

Figure 2 shows the way the median represents the ten curves in a section of the piece of music. This figure illustrates how the median represents the general tendency of the ten curves and how each one of the curves contributes to the resulting curve in the measure that it agrees with the majority of the other curves. The median for the whole piece is shown in Figure 3. The curve reflects the underlying structure of the piece: *AABA'*. *A* goes ap-

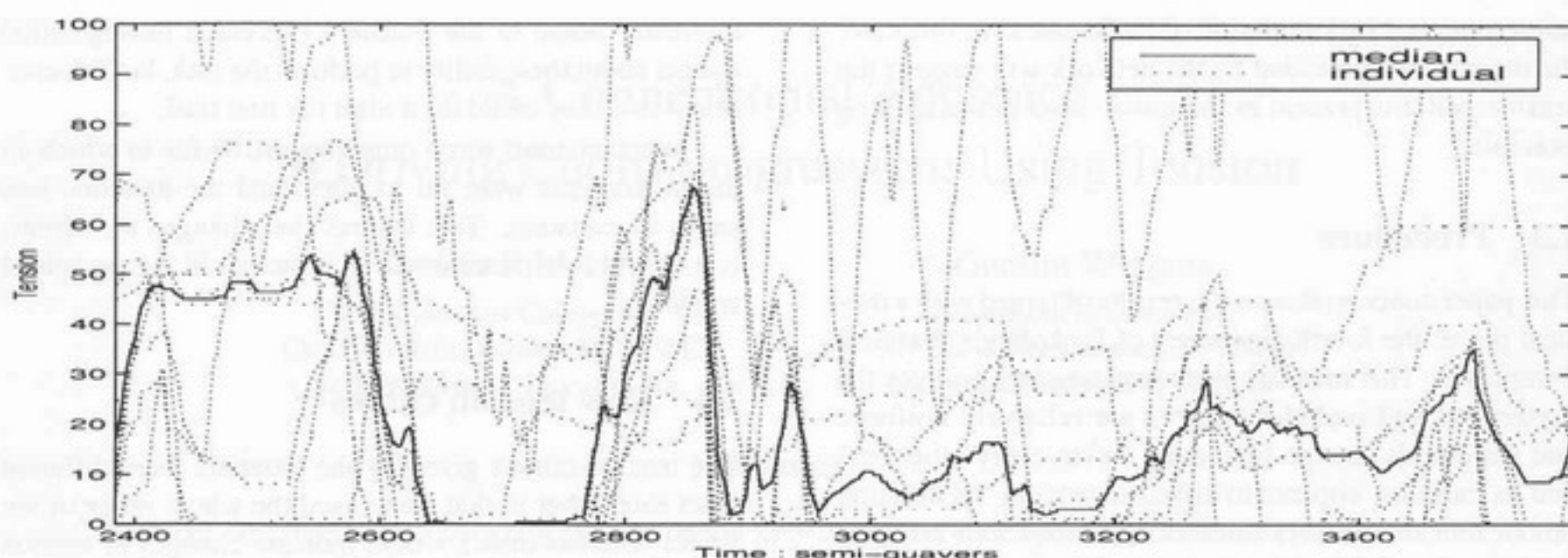


Figure 2: Ten tension curves and their median

proximately from bar 0 to bar 90, then repeats from 90 to 180, and then its variation A' begins at approximately bar 240. This and the statistical agreement in the responses of the listeners give confidence in the curve as a representative indication of the development of the piece.

3 Stage 2: Training neural networks to produce chords based on given tension curves

The concept of this system is that it should produce the chord that continues a sequence to convey a given degree of tension. The input of the neural network is thus the set of preceding chords, and the degree of tension. With the piece of music chosen, time slices of one bar are appropriate, as in most cases in it, the chords take integer numbers of bars. The set of chords used in the input is then the n chords in the preceding n bars (where n is the size of the time window).

Besides the structure of the network itself and the learning parameters, the main variables in the definition of the system were the representation of the chords and the window size. Different representations of chords and window sizes (n) were tried, to find those with which the network could produce the best results.

The chord representations tried were based on the occurrence of each one of the twelve tones in each bar. Some variations in the representation included the specific identification of the lowest tones in the bar and the consideration of the percentage of time that each tone occurred in the bar.

In each case, the architecture of the neural network was optimised. The training procedure was standard: one fifth of the data from the musical piece was chosen at random and used for testing the network (and excluded from the data used for training). Each datum contained an output chord, its average tension, and the n preceding chords.

Training data was presented to the network at random. Training stopped when the accuracy of the network on the testing data started to decrease—which means that the network is starting to memorise more than it is generalising.

3.1 Parameters of the best network built

The representation found to be the best indicated the fraction of time that a given tone would sound during the bar. Doubling was considered in the representation, so if, for example, the same tone is being played simultaneously in two different octaves, that would count twice. Each chord would thus be represented by twelve parameters, one for each of the 12 tones.

The ideal window size for this representation was of 13 bars. It was found to be ideal because the median of the error with the test data with three different random seeds was lower than for other window sizes. These two properties—especially the window size—relate directly to characteristics of the piece of music used, as will be discussed later.

3.2 Analysis of the trained network

Neural networks behave like black boxes in the sense that the knowledge is distributed rather than localised, so rules cannot be extracted by looking at the network itself. However, it is possible to gain some knowledge about the behaviour learnt by the network by 1) training different networks with different information and representations, and comparing their performance; 2) analysing the weights of the links in a trained network to find out the relative importance of the input factors; and 3) testing the trained network for different kinds of input and finding the sensitivity of the outputs. All these functions can be seen as a statistical analysis of the input data; because what the network does during learning is generalising from the training data.

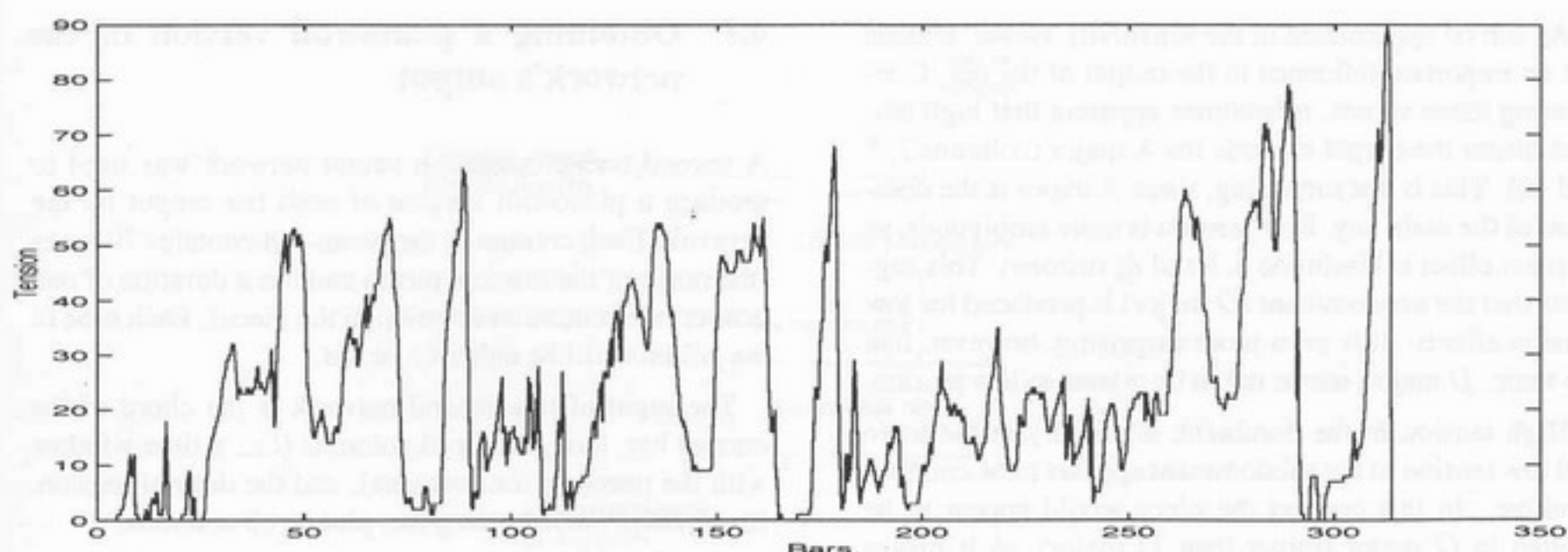


Figure 3: Tension curve for the fourth movement of Prokofiev's Classical Symphony

In this case, the training data was harmonic information from Prokofiev's Classical Symphony, and so the analysis of the properties of the network relate to this piece and do not necessarily apply to other pieces of music.

A close-to-best performance was obtained by the representation that included information about doubling, but that the addition of the bass line does not produce significant improvements. Including doubling information makes the identity of the chord clearer, as this would be expected to scale down the relative occurrence of auxiliary notes. This may also suggest that the density of the sound is a general factor that affects the chord sequence; *i.e.*, some chords sequences appear to be reserved for denser parts of the piece of music.

It is possible to perform sensitivity analysis on the network to determine the effect of changing the inputs. This was done by recording the differences in each output node after varying each input datum for each example by 5 and -5 percent of the range of values the corresponding input node can take. The results for all the examples are averaged, producing values representing the sensitivity of each output to changes in each input.

Figure 4 shows the average of the absolute values of the effects of each input chord to the output chord. It indicates the relative relevance each bar has in the chord the net outputs. The chord in the last bar (column 13 in the figure), for example, has the strongest effect. This is reasonable considering that the output is its immediate continuation, probably similar in density and often continuing the same chord. It is significant that the even numbered columns tend to have a stronger effect than their neighbours, as this suggests a harmonic rhythm.

By considering the effect of tension on each one of the tones of the output chord (Figure 5), some insight can be gained as to the way harmonic tension behaves in the piece. In the figure, column 1 relates to the sensitivity of the output c to tension, column 2 relates to $c^\sharp$, column 3 to d , and so on.

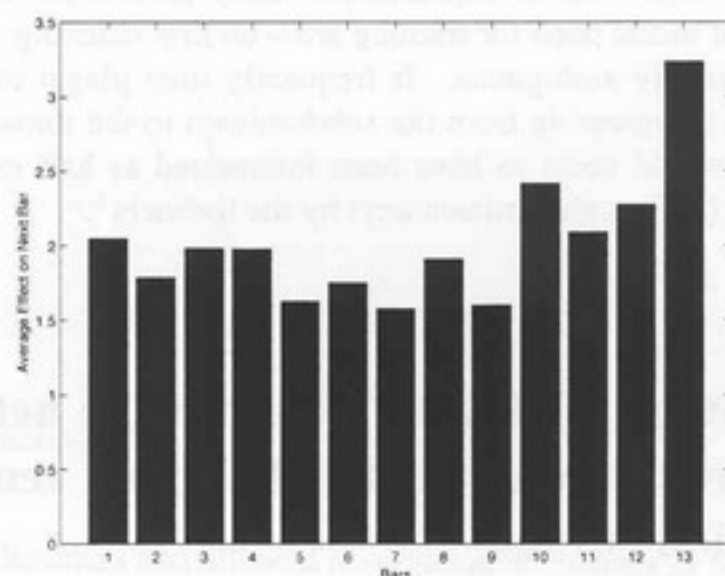


Figure 4: Average absolute sensitivity of the output chord to the input chords

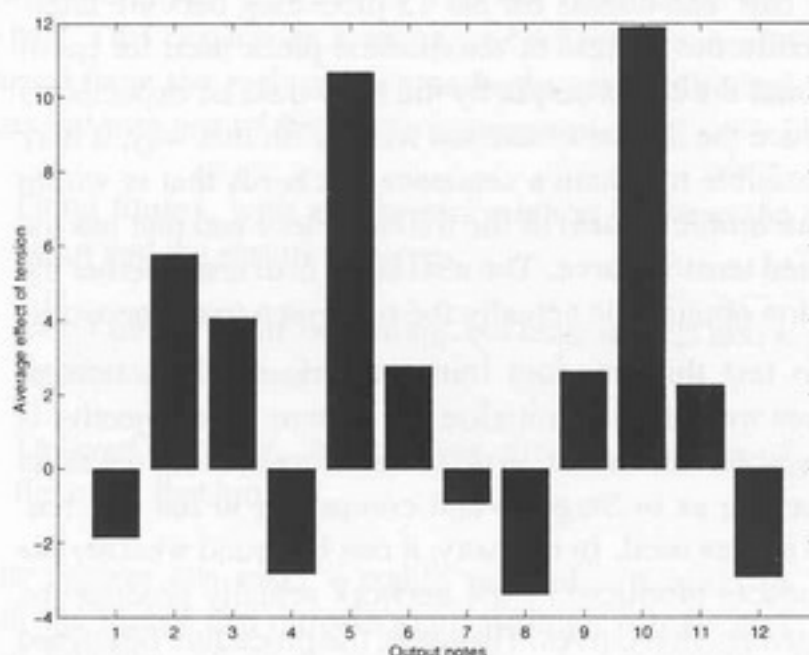


Figure 5: Sensitivity to the tones in the output chord to tension

As can be appreciated in the sensitivity values, tension has an important influence in the output of the net. Considering these values, it becomes apparent that high tension biases the output towards the A major (columns 2, 5 and 10). This is not surprising, since A major is the dominant of the main key. Low tension is more ambiguous, as its main effect is biasing to *g*, *b* and *d*♯ (tritone). This suggests that the subdominant (*G* major) is produced for low tension effects. It is even more surprising, however, that the tonic, *D* major, seems not to be related to low tension.

High tension in the dominant, ambiguity in the tonic and low tension in the subdominant appears to be counter-intuitive. In this context the piece would appear to be written in *G* major (rather than *D* major), as it makes more sense to consider that low tension is associated to the tonic, some ambiguity in tension to the dominant, and high tension to the dominant of the dominant.

This result has an explanation: many phrases of the piece of music used for training are—on first listening—harmonically ambiguous. It frequently uses plagal cadences (progressing from the subdominant to the tonic), which would seem to have been interpreted as half cadences (in the subdominant key) by the listeners¹.

4 Stage 3: Testing whether the network's output fits the given tension curves

What would be expected at this point is that by giving 13 preceding bar-chords to the network and a desired degree of tension to the network, it will output the chord for the next bar. The chords for the 13 preceding bars are interpreted in the context of the musical piece used for training, and the chord output by the net would be expected to produce the degree of tension wanted. In this way, it may be possible to obtain a sequence of chords that is within the harmonic context of the training piece and that has the wanted tension curve. The next stage is to test whether the tension obtained is actually the one given to the network.

To test this, random initial chords and two tension curves were used to initialise the system. The objective is to measure the tension in these sequences (using the same procedure as in Stage 1) and compare it to the two tension curves used. In this way, it can be found whether the sequences produced by the network actually produce the wanted tension curves. However, the procedure described in Stage 1 requires a piano-roll (or a similar representation).

4.1 Obtaining a piano-roll version of the network's output

A second backpropagation neural network was used to produce a piano-roll version of each bar output by the network. Each column in the piano-roll contains 70 tones (the range of the original piece) and has a duration of one quaver (the minimum duration in the piece). Each tone in the column will be either on or not.

The input of this second network is the chord of the current bar, four piano-roll columns (*i.e.*, a time window with the previous four quavers), and the desired tension. Its output is the following two piano-roll columns.

A small time window was chosen because the purpose of this network is to convert chords into piano-roll columns rather than to memorise melodies or the piece of music. The output contains two columns to exploit the fact that the first quaver is on the beat while the second is not.

This network was trained using data from the original piece. The output nodes represent notes in the piano-roll columns; but their values are real numbers between 0 and 1—not binary. The original piece was used to determine the thresholds for each output node that maximise the accuracy of its output.

The whole system is illustrated in Figure 6. The main network ("bar-scale net") is used at the beginning of each bar to generate the chord (the "bar suggestion") that will produce the degree of tension in the input. The piano-roll network is used every two quavers to produce the following two piano-roll columns ("next two quavers"), based on the chord, the four previous piano-roll columns and the tension.

At the end of the bar, after eight piano-roll columns have been generated, the chord equivalent to these eight columns (rather than the chord output by the main network) is used to update the time window at the input of the main network. This is to allow for the fact that the piano-roll network may distort the chords in the main network output.

For testing purposes, two sets of pieces were created using this system. All sound sequences were under 80 seconds long. The first set consisted of two pieces generated to produce a section of the tension curve of the original piece. The second set consisted of eight pieces generated using a skewed sinusoidal wave as tension curve. The chord and piano-roll windows were initialised with random data for each one of the pieces. Importantly, the random seeds and the resulting sound sequences were not evaluated or selected before the experiment.

As expected, the piano-rolls produced by the system described are not melodic, and listeners expressed that they sounded almost like random music, while admitting that some "order" and even "style" could be noticed in them.

¹This ambiguity was reportedly introduced intentionally, as it was frequently exploited by Haydn, who was one of the models Prokofiev used for the Classical Symphony.

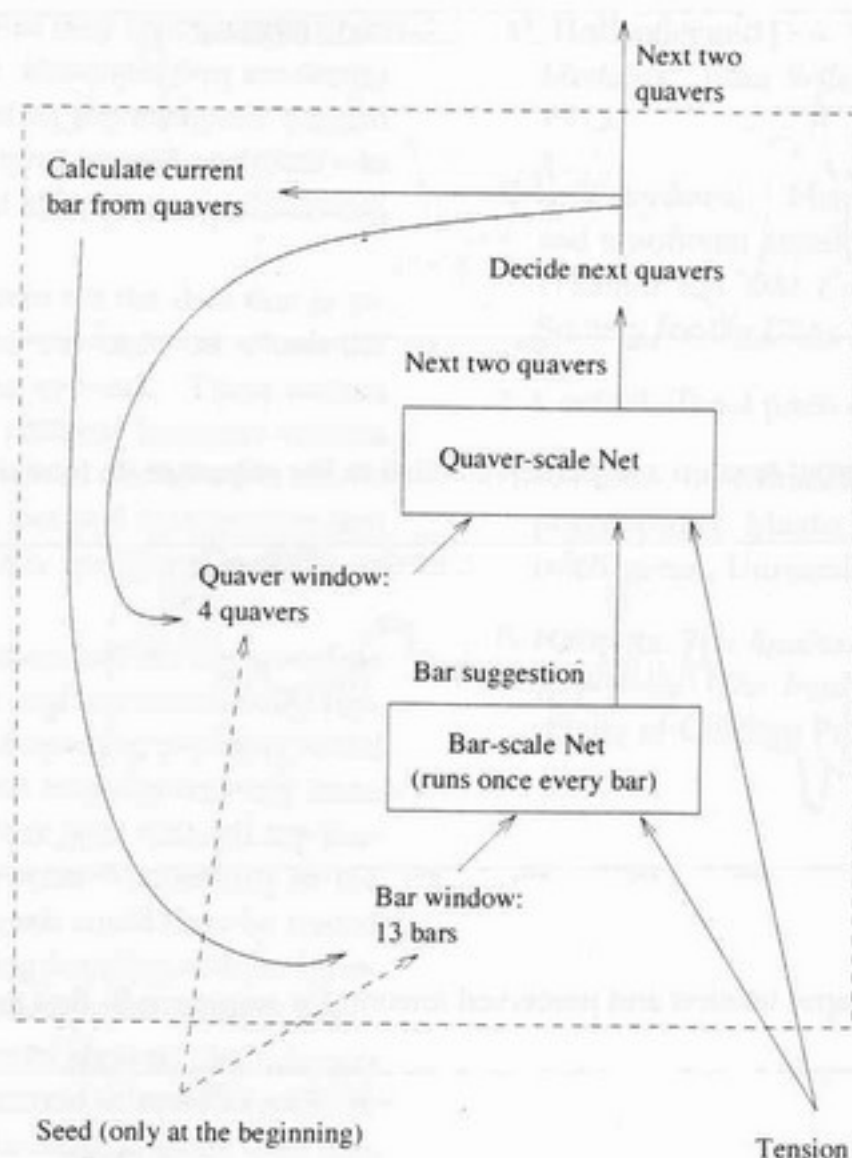


Figure 6: System to produce piano-rolls

4.2 Test procedure

At this stage, 16 Western subjects listened to the original piece of music and recorded the tension they perceived using the sprung wheel — both to let them get used to the experimental settings and to find differences between their response and the original tension curve used to train the network. After this, each participant recorded the tension for the two pieces in the first set and for four of the pieces in the second set. Listeners heard the sound sequences generated by the system in random order.

The data obtained was normalised and processed in the same way as in Stage 1, scaling the individual curves from 0 to 100 and extracting the median.

4.3 Results and analysis

Figures 7, 8 and 9 illustrate the results of Stage 3. Figure 7 shows the tension curve obtained for sequence A of the first set. In this case, the fitness of the resulting curve to the input tension is quite clear.

Figure 8 shows the tension curve obtained for sequence B of the first set. In this case, even though at some points the resulting curve moves in the same direction as the input tension, their fitness is poor. The problem in this case is that the random seed used to initialise the piano-roll produced very dense columns which were uncomfortable

to the listeners and allowed no noticeable changes in the chords or piano-roll.

Figure 9 shows the input tension curve used for the second set and the average of the tension curves obtained for its eight sequences. This curve has a good correlation in the first half and a tendency to a steady state in the second half. This curve is an average, and reflects the results obtained from the eight sequences in the set. Individual pieces fall into one of the following categories:

- Good fitness, with an obvious relation between the input and the resulting curves.
- Good fitness at the beginning, but tension falls into a steady state by the end.
- Delayed response: the resulting curve is an offset of the input tension.

The system can enter a stable state of saturation in which the inputs and outputs maintain high values, and this clearly depends on the random data used to initialise the input piano roll columns and chords.

4.4 Statistical analysis of the results

To test the fitness of the tension curves, they were compared to the tension curves used to generate the sound

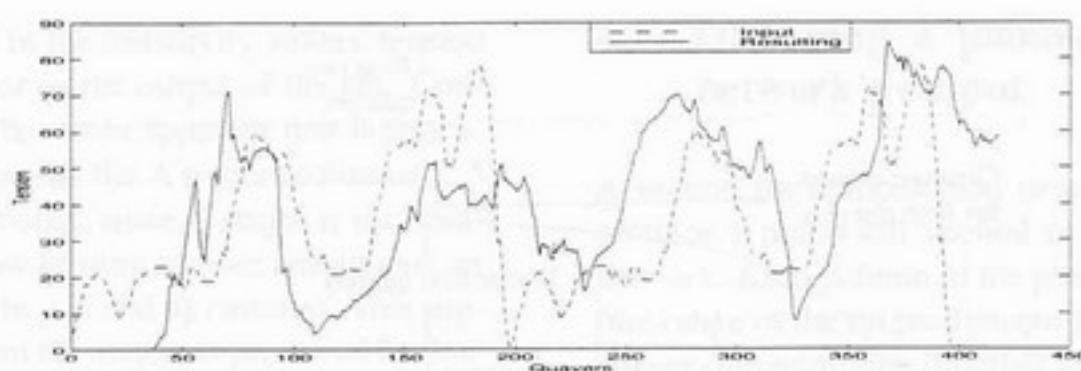


Figure 7: Input tension and perceived tension for sequence A, first set

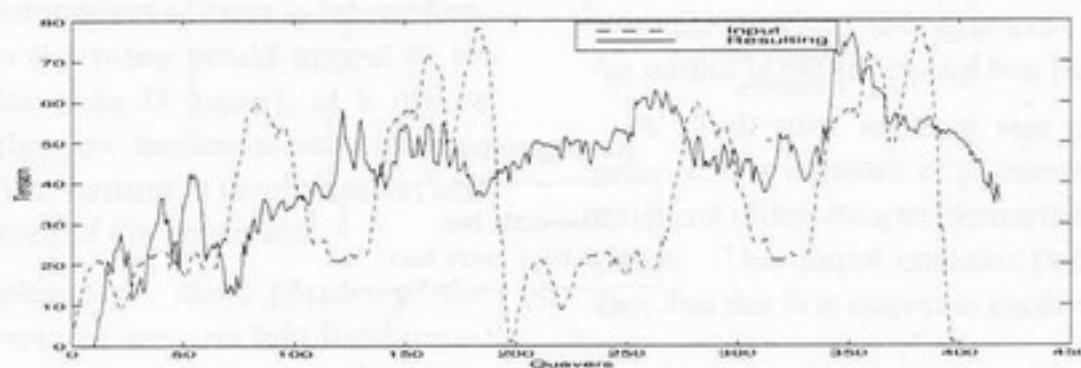


Figure 8: Input tension and perceived tension for sequence B, first set

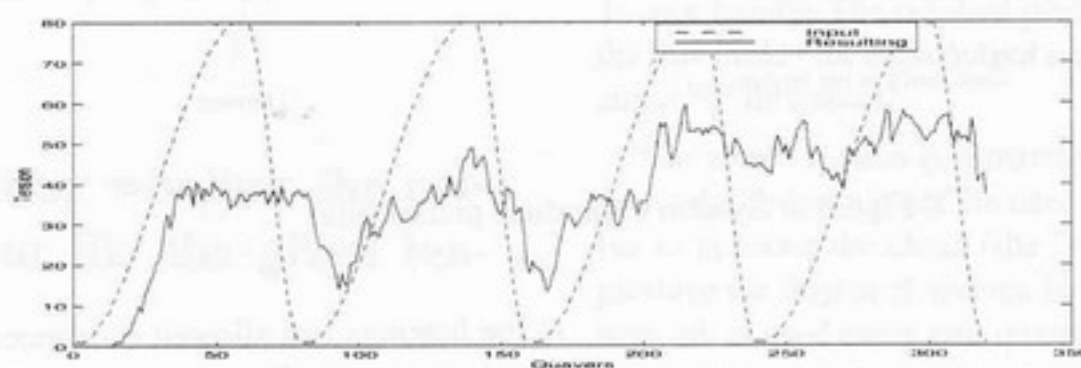


Figure 9: Input tension and average perceived tension for the second set

sequences. The comparison was done using the test of rank correlation (which is equivalent to applying Friedman's test to two data series only). The choice of a non-parametric test is once again justified by the fact that a distribution of tension cannot be assumed.

The rank correlation between the input tension curves and the tension indicated by the listeners was statistically significant in all the cases, but suspected not to be as strong as that found among the listeners in the original test.

The tension curves recorded for the original piece of music can be used as reference to test whether these differences could have originated from the different set of listeners that participated. The procedure for this comparison involves calculating the rank correlation for the tension curves obtained in Stage 1 and Stage 3 for the original piece. This value is then compared to each one of the correlations, considering the degrees of freedom in the data series. The results indicate that the correlation between the two sets of listeners is better by far—so the perception of tension of the listeners that participated in

Stage 3 is similar enough to that of the listeners in Stage 1.

In most cases, the correlation between the input tension and the perceived tension has high statistical significance nonetheless. The results are encouraging in that they show that the network learnt to reproduce aspects of the harmony to produce a given tension. The network interpreted the random input data in terms of the harmonic context it learnt, and produced the sound sequence based on it. In some cases, however, the initial random data forced the system into a stable state of saturation.

5 Discussion

The backpropagation network appears to have learned patterns that exist between harmonic information and the tension measured, within the harmonic context of the musical piece used. In some cases it is able to produce sound sequences that produced responses in the users equivalent to those measured for the original musical piece. These

“successful” sequences are new as they are clearly distinct from the original musical piece. However, they are copies in the sense that they are based on the harmonic context of the original. What was not learnt from the original was the result of random factors that affect the neural network or of external inputs.

The random factors in this case are the data that is removed for testing purposes and the order in which the training data is presented to the network. These factors affect the way it contains the patterns harmony-tension patterns. Once training has finished, the network can be considered to be deterministic, but still its structure and outcomes will be affected by the random factors in its training.

The chords used for initialisation and the input tension curve are inputs to the system and not necessarily random. They affect the resulting sequence, probably at the same level that the musical piece used for training does, as they represent the material that goes through the harmonic context learnt by the network. The novelty of the sequence produced by the network could thus be traced back to the random factors during learning and the interaction of the system inputs and the systems complexity.

At a more basic level, the results showed that listeners tend to agree in what they understood as musical tension. The tension they perceived in the musical piece used was found by the neural network to correlate to the harmony, to the point that it could produce chords appropriate for the context and desired tension. The resulting structure of the backpropagation network reflects important features of the way harmony is perceived by listeners. The results ultimately suggest that the concept of tension can be used to drive the production of musical sequences.

References

- J. Berger and D. Gang. A neural network model of metric perception and cognition in the audition of functional tonal music. In *International Conference for Music Perception and Cognition*, pages 23–26. International Computer Music Association and Program of Psychoacoustics of IPSA, ICMA and IPSA, 1997.
- J.J. Bharucha and P.M. Todd. Modeling the perception of tonal structure with neural nets. In P.M. Todd and D.G. Loy, editors, *Music and Connectionism*, pages 128–137. The MIT Press, Cambridge, MA, USA, 1989.
- K.A. Brownlee. *Statistical Theory and Methodology in Science and Engineering*. John Wiley & Sons, New York, NY, USA, 1965.
- M. Dolson. Machine tongues xii: Neural networks. In P.M. Todd and D.G. Loy, editors, *Music and Connectionism*, pages 3–19. The MIT Press, Cambridge, MA, USA, 1989.

- M. Hollander and D.A. Wolfe. *Non-parametric Statistical Methods*. John Wiley & Sons, New York, NY, USA, 1973.
- C.L. Krumhansl. Musical tension: cognitive, motional and emotional aspects. In *Proceedings from the Third Triennial ESCOM Conference*, pages 3–12. European Society for the Cognitive Sciences of Music, 1997.
- F. Lerdhal. Tonal pitch space. *Music Perception*, 5, 1988.
- A.F. Melo. A connectionist model of tension in chord progressions. Master’s thesis, Department of Artificial Intelligence, University of Edinburgh, 1998.
- E. Narmour. *The Analysis and Cognition of Basic Melodic Structures: The Implication-Realisation Model*. University of Chicago Press, Chicago, IL, USA, 1990.

A n-gram approach to fugal exposition composition*

Alastair Craft*

*Department of Computing, City University
Northampton Square, London, EC1V 0HB
cp456@soi.city.ac.uk

Ian Cross†

†Faculty of Music, University of Cambridge
11 West Road, Cambridge, CD3 9DP
ic108@cam.ac.uk

Abstract

An algorithm is presented that uses a multiple n-gram method to learn and compose fugal expositions from a training corpus. The method of analysis used by the algorithm is described and sample output is given with evaluation.

1 Introduction

This project set out to apply n-gram techniques to the composition of counterpoint, in particular the composition of fugal exposition. N-grams are a probabilistic language model that originate in natural language processing. They are a simple model where N-grams and Markov language models have previously been used to model music, usually melody, for instance, Pinkerton (1956) used a high-order Markov model to generate nursery rhymes, and both Jr. et al. (1993) and Conklin and Witten (1995) used models for generation of chorale melodies. Ponsford et al. (1999) and Barbaud (1966) have both used n-grams for modeling harmony, while the work of Cope (1991) uses n-gram based models for stylistic replication of composers work and the work of Pachet (2002) uses a Markov model of musical styles to provide a real-time interactive system for improvisation.

Both Conklin and Witten (1995) and Ponsford et al. (1999) note the success of the models that are used for creation of original and acceptable strings and the inference of harmonic framework using n-grams. These successes in different areas of music suggest that the individual approaches can be combined to create a framework for creating counterpoint, a type of music where musical lines sound simultaneously according to a series of rules. However, counterpoint raises some specific problems for n-grams: they are unable to replicate long-distance dependencies and they are unable to model more than one string at a time, unless those strings are encoded together in some way. Both of these are problematic for compositional processes, except when they can easily be modeled linearly (such as harmony or melody), but are particularly problematic for counterpoint, where the individual lines are to some extent independent and material does have long-distance dependencies, both in time and between voices. Indeed, Cope admits that "contrapuntal music poses serious problems to machine composition" Cope (1991) and his attempts to use EMI (Cope (1996)) to

create counterpoint depend upon special programs that replicate structure - his system cannot inherit contrapuntal forms in the same way as homophonic forms.

The current work sets out to produce a framework where the problems of dimensionality and long-distance dependency can be constrained and dealt with by using multiple n-grams to describe a contrapuntal work, and by the construction of a lexicon specific to a piece to deal with re-use of motivic material.

2 Implementation

2.1 Operation of algorithm

For the purposes of the model, for an exposition with n voices, $2n + 4$ different n-gram representations are necessary. 4 of these are global to the piece: the harmony, the structure, the subject-contour and the subject rhythm. The remaining $2n$ n-gram representations are specific to each voice and consist of voice-contours and voice-rhythms: the overall contour of each voice and the local diminution of the voice are dealt with separately.

2.2 Structure of pieces

Whilst both the harmony and the individual voices can be recreated using n-grams, there are dependencies both *between* voices, resultant from imitative counterpoint, and within any voice, resultant from the re-use of material. Some hybridised n-gram method that can consider an item *greater* than a note *when appropriate* could overcome this problem. The length of this variable unit is defined and constrained by a structural description of the corpus: by combining a pattern-matching and n-gram process a description is achieved that describes the structure of the exposition (based upon the motivic material used) and this is used to constrain the composition process.

Huron (1989) demonstrates that it is difficult for a listener (including trained musicians) to correctly identify

*This work was carried out whilst A Craft was at the Computer Laboratory, Cambridge University.

an inner entry¹, or even to identify it at all, when listening to contrapuntal writing and that as the number of concurrent voices in the texture increases, participants were slower to identify the addition of new voices. Huron and Fantini (1989) went on to demonstrate that fugal writing reflects this in the cases where the number of voices is high. This suggests that simple rules determining the introduction and decrementation of voices in a piece can be modeled in some simple way, given a description of the entries and exits of voices.

2.3 Harmonic analysis

2.3.1 Operation of harmonic analysis

The harmonic analysis algorithm used is a modified version of Temperley (1998), which bears closest resemblance to Pardo and Birmingham (2002), in that it uses chord templates rather than relations to chord roots, as Temperley's algorithm does. The algorithm produces a description of the harmony of each piece, and these are used to produce probabilities for 4-gram generation of new harmonic frameworks for fugues. In addition to the progressions stored, the lists of spellings of notes found in the source data are recorded.

2.3.2 Chord-spacing and voice leading

Whilst a 4-gram is used to construct the harmonic framework of a piece, it alone is not sufficient to create full counterpoint other than first species, and even then with the risk of violating standard contrapuntal rules. Information on chord spacing (which notes occur where and at what relative height in a chord) and voice leading (possible movements from one chord to the next) for this to be remedied. Tables are constructed from the corpus of possible spacings of chords and possible chord transitions.

2.4 String analysis and string matching

The string analysis and string matching algorithm results in two pieces of information per exposition from the corpus: the fugue subject and a structural description.

Before the string matching takes place proper, there is a pre-processing stage, where the diminutions of the *tactus*² starting on each beat are removed, the strings to be analysed are repopulated with a reference to the diminution used at that point, its starting pitch and its duration. If there is an item that is greater in length than the beat (such as ties, or longer-duration notes), then diminutions are removed in whole number multiples of the beat. Diminutions that have the same scale-degree movement (that is, that match for their entire length their name codes), but have differing pitch codes are recorded as variations of

the same diminution. In addition to the diminution, two additional pieces of information are stored: the entrance and exit vector, being the directions by which a diminution is approached and exited and which fall into six categories: leap-wise above, where it is approached or exited by greater than a scale degree; step-wise above, where it is approached or exited by a scale degree; step-wise below; leap-wise below; same-height, where the diminution is approached or exited on the same scale degree as it begins or ends; and none, which occurs if the diminution is preceded or followed by a rest, or start or end of file marker.

The pattern matching then operates such that any sub-strings seen as matches satisfy one of the following conditions:

Perfect match For the entire length of the sub-strings seen as matches, S^0 to S^{n-1} , the name-code intervals between any pair of adjacent events S_i^j and S_{i+1}^j are the same and the diminution of each event S_i^j is identical, for all j .

Special case 1 At some point, p , in all sub-strings, $S_{0,\dots,p}^j$ and $S_{p+1,\dots,end}^j$ are seen as perfect matches for all j and the difference in name-code intervals between events S_p^j and S_{p+1}^j for all j is no greater than one.

Special case 2 At some point, p , in all sub-strings, $S_{0,\dots,p}^j$ and $S_{p+2,\dots,end}^j$ are seen as perfect matches for all j and the rhythm of the diminution at S_{p+1}^j is the same for all j , and the difference in name-code intervals between events S_p^j and S_{p+2}^j for all j is no greater than one.

Each piece is then described in terms of the repeating patterns that occur in it, their duration, the voice that they occur in, their starting pitch and their starting time. A full description of the operation of the pattern matching algorithm, and of the entire algorithm, is in Craft (2002).

2.5 N-grams for each voice

As noted in 2.4 above, each voice is reduced to its *contour* and *diminution*. Probabilities are calculated for a 3-gram description of the contour and for the diminution, probabilities are calculated for a 2-gram description *based upon the rhythm of the diminutions alone*. Any maximally repeating material is excluded in this general description, but is used to calculate probabilities for the subject in the same way.

2.6 N-gram regeneration of pieces

For an exposition with n voices, $2n + 4$ different n-gram representations are necessary. 4 of these are global to the piece: the harmony, the structure, the subject contour and

¹That is an entry of a fugue subject where the entry occurs in an inner-sounding voice, that is that there are voices currently sounding both above and below it.

²Diminution in its ornamentation sense.

the subject rhythm. The remaining $2n$ n -gram representations are specific to each voice and consist of voice contours and voice-rhythms.

The algorithm for creating a new fugue is given in Figure 1.

3 Evaluation

3.1 General Evaluation Constraints

If the model presented in this algorithm is accurate, the expositions should be typical of Bach-style fugues, major and minor, three-part and four-part. As elements of the algorithm, and ideas about music generally are not well-formed there can be no absolute way to demonstrate the correctness of the algorithm. However, it is still possible to evaluate the efficacy of the algorithm by testing the output against a set of questions or aims. Questions to be answered about the generated pieces are:

- are subject/answer progressions learned
- are both tonal and real answers learned
- are any contrapuntal rules violated
- is the counterpoint produced idiomatic
- are harmonic progressions learned
- are cadences learned

3.2 Discussion of Sample Output

Sample fugue expositions generated by the algorithm are available from www.soi.city.ac.uk/~cp456/fugue.htm.

The 4-part fugal exposition, created using seven fugal expositions³ from *Das Wohltemperierte Klavier* Bach (1989) Bach (1995) and 3-part fugal exposition, created using five fugal expositions⁴ from *Das Wohltemperierte Klavier*

In all the expositions generated by the algorithm, the standard pattern of subject followed by answer was followed. However, due to the simplicity of the pattern matching algorithm, which sees both tonal and real answers as the same, the algorithm can currently only produce real answers. This is of no matter to the resultant new fugues if the structural model generated for re-composition allows for modulation of the piece back to the tonic for the third entry, as happens in the 4-part fugue, but if, as in the 3-part fugue, such a modulation does not occur, an abrupt harmonic change happens. It is this inability to replicate tonal answers that partly causes the

harmonic ambiguity that is in this fugue, but the choice of an inappropriate starting figure does not help.

None of the fugues generated by the algorithm violated any rules of counterpoint - this is due to the constraint put upon the output by only being able to re-generate transpositions of voice-leading that it has already seen. The counterpoint produced by the algorithm is variable. Consider 4⁴ 5 and bar 5 of the 4-part fugue (Figure 2), which demonstrates the variability of the counterpoint created. 4⁴ is considered bad, as all of the voices are still for an entire beat, but such moments are generally rare. Bar 5, on the other hand is an example of reasonably good counterpoint,

with decent contours to the lines and maintenance of rhythm. Harmonic progressions used are good, and cadences created (usually at the end of the exposition) are also idiomatic.

Whilst the resultant fugues are reasonable, the contours of their subjects could be argued to follow too closely those of the source data - the 4-part fugue follows the contour of BWV 850, although it populates the subject with sufficiently different motif-leaves that the contour-similarity does not make the two seem too similar.

One issue that is of interest, as not representation for it was coded explicitly, is the generation of sequences by the algorithm. Consider 7² to 8⁴ of the 3-part fugue (Figure 3). Because the subject is a sequence at that point, and because already-used motif-leaves are used in preference to those not yet used, the sequential nature of the passage is replicated in all parts by the algorithm.

3.3 Further Issues

Attempts at re-creation of structures larger than the exposition are less well dealt with by the algorithm - this is partly due to the time constraints in preparing source data (which takes upwards of a day per fugue) meaning that no real representative source data of a fugue as a whole could be constructed, as no sufficiently large corpora was presented to the algorithm, leading to verbatim replication of the source structures, and partly due to repetition: some passages, being statistically more likely than others tended to be re-composed more than once, leading to uninteresting fugues. This implies that the unification of structures is not altogether capable in its current guise for replication of fugues. However, being able to replicate a fugal exposition does demonstrate the ability of this algorithm to perform the task that is currently unfulfilled by Cope's *EMI*, namely to create imitative counterpoint using recombination without recourse to special code.

One problem that has not been able to be explored, is the problem of backtracking in the algorithm. As explained in Section 2.6, the algorithm builds the lexicon of repeated patterns in a left-to-right manner, with backtracking to the point at which a particular pattern first occurs to modify it. Whilst creating relatively short pieces

³The expositions used were taken from BWV 846, BWV 850, BWV 862, BWV 868, BWV 574, BWV 877 and BWV 886.

⁴The expositions used were taken from BWV 847, BWV 853, BWV 875, BWV 879 and BWV 883.

⁵Bar 4, beat 4

1. Generation of structural element 0

- (a) A *key-signature* and a *time-signature* are chosen at random from the input pieces.
- (b) An initial structural unit is chosen, using the n-gram method described above.
- (c) A second structural unit is chosen, using the n-gram method described above.
- (d) Any labelled pattern that occurs in the first structural element is constructed. As this is always the subject, the rules for subject construction are used to generate a subject-contour. Any subject contour that does not terminate at the correct position in the bar is rejected.
- (e) The rhythm of the motif-leaves to populate the subject-contour are chosen, using the n-gram method described above.
- (f) Motif-leaves are chosen that match the rhythm chosen in step 1e. These are then used to populate the subject-contour, using the entrance and exit vectors to ensure good-shape in the melodic line. Inversions of motif-leaves are allowed, resulting in a larger lexicon of motif-leaves. If a motif-leaf is inverted, so are its entrance and exit vectors. If no solution can be found, the algorithm backtracks, and attempts with a different subject-contour, starting again at step 1d. The subject, being a repeated pattern is then additionally stored in a separate list, or *lexicon* of repeated patterns. The motif-leaves used in populating the subject-contour are also stored in another separate list, and are used in preference over motif-leaves as yet unused in composing later passages.
- (g) A harmonic analysis of the subject is taken. This is checked against the information stored about spellings of notes given a harmonic context, and spellings of notes are adjusted as necessary.

2. Generation of structural element $n + 1$

- (a) Any patterns that are determined by the structural unit $n + 1$ are populated with corresponding elements from the lexicon.
- (b) The patterns are checked for correct progressions. If invalid, the algorithm backtracks to the point at which the most recent pattern was generated and re-composes it at that point.
- (c) Generation of new patterns
 - i. The harmonic analysis taken in 1g is presented to the harmony n-gram representation to find possible next harmonies. Only possible next harmonies which include the starting pitch of the answer and the starting pitch of the counter-subject as a harmony note are chosen, and one of these is chosen at random.
 - ii. A duration for the starting pitch of the new pattern is chosen from the n-gram method for regeneration of string contours. It is then populated with a motif-leaf of the same duration, which is chosen so as to conform to the restraints of the chord-spacing, voice-leading information as well as the exit vector of the end of the subject.
 - iii. The spellings of the notes in the motif-leaf of the first event of the new pattern are modified to conform to the harmony.
 - iv. A harmonic analysis is taken of the piece up to the point so far completed, to find possible next harmonies. Only possible next harmonies which include the pitch in the answer at the current point are chosen.
 - v. The n-gram method for regeneration of melodic lines is used to suggest possible next notes for the pattern. The first which corresponds to the harmonic analysis is accepted.
 - vi. Steps 2(c)i to 2(c)v are repeated until the pattern is entirely populated. If at some point, $i + 1$, no solution can be found, then the algorithm backtracks to point i and tries a different possible solution. If this too fails, the algorithm backtracks to point $i - 1$ and so on.
- (d) Generation of free parts
 - i. Steps 2(c)i to 2(c)v are repeated for all other parts that are active in structural unit $n + 1$. If no solution can be found the algorithm backtracks to modify the patterns that cannot be changed locally.

Figure 1: The compose algorithm



Figure 2: Bars 4 and 5 of the 4-part fugue



Figure 3: Bars 7 and 8 of the 3-part fugue

(which an exposition is), this creates no real problem, but if the algorithm was used in its current guise for generation of a whole fugue, this could rise to wasted composition. An example where this could be utterly wasteful is the composition of *stretto*, where the voices enter with the fugue subject, but the entries overlap, so that the fugue subject is played against itself. Strettos usually occur right at the end of the fugue, and if all the input fugues had stretto, this would be expected in the output fugues. However, if the subject chosen is not suitable for stretto, there would be vast amounts of backtracking - in effect all the composition done up to that point would be wasted.

4 Conclusions

Whilst the results achieved by this algorithm represent an advance on the results achieved by *EMI*, there are still a number of areas in which the algorithm could be improved. The harmonic-analysis component, whilst not producing results that would be produced by an expert given the same task are sufficient for the purposes of the algorithm. The two major areas that could be refined are the string-matching and the structural analysis and unification components.

4.1 String Matching and Structural Analysis

Currently, the string matching component will not find all possible matches. In particular, matches that would be

found by a music analyst, such as *diminution*⁶ (reducing the length of notes in a passage by a constant, usually making all notes have half the original duration), *augmentation* (increasing the length of notes in a passage by a constant, usually making all notes have twice the original duration) and *inversion* are not found; neither are *false entries* (an entry that begins with the subject, but diverges from it some way along its length). These are all techniques that are commonly used to transform a fugue-subject for use later on in the piece. Also, the failure of the algorithm to replicate larger structural units than just the exposition implies that either n-gram modelling of musical structure is not suitable, except over short periods, or that the unification of differing analyses is not sufficiently capable. Possible future work on the algorithm could introduce ways of deriving more general structural analyses of music.

4.2 Re-composition

The backtracking problem could, at best, be wasteful and, at worst, possibly prevent the composition of new pieces. This suggests that the structural framework for a piece should be calculated in its entirety and then parsed for potential problem moments, that is moments which have the most number of constraints in terms of motivic material and that these are composed first, in effect, filling the lexicon for a piece as the first stage of composition, rather

⁶Rather confusingly, the word *diminution* gets used for a number of different purposes by musicologists.

than creating the lexicon as the composition progresses. This would help to eliminate substantial backtracking that is endemic in the algorithm as it currently stands.

4.3 Chord Rhythm

As demonstrated in the sample output (Section 3.2), at some points the dynamism of the piece is destroyed by a lack of rhythmic movement at a particular point. This could be remedied by another global constraint that keeps track of the rate at which events happen.

4.4 Future Directions

The current algorithm requires multiple parses of the input data in order to create the various representations of fugue to generate new pieces. Ideally, an integration of the harmonic-analysis and string-analysis components should be developed and used, although how these should be integrated is less clear, as they would require some single-string representation of counterpoint, or alternatively, would require some representation that did not use strings, such as the multidimensional dataset representation that is used in Meredith et al. (2002).

The harmonic-analysis component is weighted strongly towards common-practice period music, which would make it entirely inappropriate for some fugues written after c.1900. Unlike the original on which it is based, however, it is possible to define the 'stable' chords, and indeed it is possible with further modification that these chords can be self-derived. This suggests that it may be possible to convert the algorithm so that it 'learns' the harmonic language of the source data, making this element extensible.

Further extensions to the algorithm would involve generalising it, so that it can replicate other structures other than imitative counterpoint. This would involve adding a classification element that determined the similarity between input pieces and used information from similar pieces to create new ones.

References

- J. S. Bach. *Das Wohltemperierte Klavier I*. Bärenreiter Kassel, 1989.
- J. S. Bach. *Das Wohltemperierte Klavier II*. Bärenreiter Kassel, 1995.
- P Barbaud. *Initiation à la composition musicale automatique*. Dunod, Paris, 1966.
- D Conklin and I Witten. Multiple-viewpoint systems for music prediction. *Interface*, 24:51–73, 1995.
- David Cope. *Computers and Musical Style*. A-R Editions, 1991.
- David Cope. *Experiments in Musical Intelligence*. A-R Editions, 1996.
- Alastair Craft. A contrapuntal algorithmic composer for fugue. Dissertation for Diploma in Computer Science, University of Cambridge, 2002.
- David Huron. Voice denumerability in polyphonic music of homogeneous timbres. *Music Perception*, 6(4):361–382, 1989.
- David Huron and Deborah A Fantini. The avoidance of inner-voice entries: Perceptual evidence and musical practice. *Music Perception*, 7(1):43–38, 1989.
- F Brooks Jr., A Hopkins Jr., P Neumann, and W Wright. An experiment in musical composition. In Stephan M. Schwanauer and David A. Levitt, editors, *Machine Models of Music*, pages 23–40. MIT Press, 1993.
- David Meredith, Kjell Lemström, and Geraint A. Wiggins. Algorithms for discovering repeated patterns in multidimensional music representations. Submitted to *Journal of New Music Research*, 2002.
- Francois Pachet. Interacting with a musical learning system: the continuator. In A. Smaill C. Anagnostopoulou, M. Ferrand, editor, *Music and Artificial Intelligence*, volume 2445 of *Lecture Notes in Artificial Intelligence*, pages 119–132. Springer Verlag, 2002.
- Bryan Pardo and William P. Birmingham. Algorithms for chordal analysis. *Computer Music Journal*, 26(2), 2002.
- R Pinkerton. Information theory and melody. *Scientific American*, 12(194):76–86, 1956.
- Dan Ponsford, Geraint Wiggins, and Chris Mellish. Statistical learning of harmonic movement. *Journal of New Musical Research*, 1999.
- David Temperley. An algorithm for harmonic analysis. *Music Perception*, 15:31–68, 1998.

An Empirical Comparison of the Performance of PPM Variants on a Prediction Task with Monophonic Music

Marcus Pearce*

* Department of Computing, City University,
Northampton Square, London EC1V OHB
m.t.pearce@city.ac.uk

Geraint Wiggins†

† Department of Computing, City University,
Northampton Square, London EC1V OHB
geraint@city.ac.uk

Abstract

N-gram models have been employed for a number of musical tasks including the development of practical applications providing computational support for creative individuals as well as theoretical studies of creative processes. Our goal in this research is to evaluate, in an application independent manner, some recent techniques for improving the performance on monophonic music of a subclass of such models based on the Prediction by Partial Match (PPM) algorithm. These techniques include the use of escape method C, interpolated smoothing and unbounded orders. We have applied these techniques incrementally to eight melodic datasets using cross entropy computed by 10-fold cross-validation on each dataset as our performance metric. The results demonstrate statistically significant performance improvements afforded by the use of all three techniques. We discuss these findings in terms of previous research carried out in the field of data compression and with natural language and music corpora and present some directions for future research. It is our hope that these improvements may be applied usefully to specific musical tasks.

1 Introduction

N-gram models have been employed for a number of research tasks in music including the development of practical applications providing computational support for creative individuals as well as theoretical studies of creative processes. In the former category, we cite models for computer-assisted composition (Ames, 1989; Assayag et al., 1999; Hall & Smith, 1996), automatic improvisation (Lartillot et al., 2001) and music information retrieval (Pickens et al., 2002); in the latter category, *n*-gram models have been used for stylistic analysis of music (Conklin & Witten, 1995; Dubnov et al., 1998; Ponsford et al., 1999) and cognitive modelling of music perception (Ferland et al., 2002; Reis, 1999).

Our goal here is to investigate the performance of a subset of such models on a range of monophonic music data in an application independent manner. We are concerned, in particular, with the application to music data of a particular technique for combining the predictions of *n*-gram models called *Prediction by Partial Match* (PPM – Cleary & Witten, 1984) which forms the central component in some of the best performing data compression algorithms currently available (Bunton, 1997). PPM has previously been applied to natural language data (Chen & Goodman, 1999) and to music data (Conklin & Witten, 1995). Since its introduction, a great deal of research has focused on improving the compression performance of PPM models and our specific aim is to evaluate the performance of these improved models on a range of monophonic music. It is our hope that these improvements, evaluated here in

an application independent manner, may then be applied usefully to some of the specific musical tasks cited above.

The paper is organised as follows. In §2, we introduce *n*-gram modelling in general and the PPM scheme in particular, as well as the information-theoretic performance metrics we shall use. Much of the background for this research is drawn from the fields of statistical language modelling (Manning & Schütze, 1999) and text compression (Bell et al., 1990). We hope to demonstrate that practical techniques and methodologies from these fields can be usefully applied in the modelling of music. As noted above, *n*-gram models have been applied to a number of musical tasks and in §3, we discuss research in the musical domain which uses related models and methodologies. The data and experimental methodology employed are described in §4. The results of our experiments are presented in §5, discussed in §6 and, in §7, we conclude by presenting a number of useful directions for future research.

2 Background

2.1 *N*-gram Models

For the purpose of describing this research we shall characterise the acquisition of knowledge about melodic music as a sequence learning problem (Dietterich & Michalski, 1986). The objects of interest are sequences of events where each event consists of a finite set of attributes and each attribute may assume a value drawn from some fi-

$$p(e_i|e_{i-n+1}^{j-1}) = \begin{cases} \alpha(e_i|e_{i-n+1}^{j-1}) & \text{if } c(e_i|e_{i-n+1}^{j-1}) > 0 \\ \gamma(e_{i-n+1}^{j-1})p(e_i|e_{i-n+2}^{j-1}) & \text{if } c(e_i|e_{i-n+1}^{j-1}) = 0 \end{cases} \quad (1)$$

nite alphabet ξ . Here our events are musical notes as notated on a score and we restrict ourselves to the attribute of chromatic pitch. We shall use the notation $e_i^j \in \xi^*$ to denote a sequence of events $e_i \dots e_j$ where $i, j \in \mathbb{N}^+$ and ξ^* denotes the set of all sequences composed of members of ξ including the empty sequence ϵ . The goal of sequence learning is to derive from example sequences a model which estimates the probability function $p(e_i|e_{i-1}^{j-1})$.

If we make the assumption that the probability of the next event depends only on the previous $n-1$ events, for some $n \in \mathbb{N}^+$:

$$p(e_i|e_{i-1}^{j-1}) \approx p(e_i|e_{i-n+1}^{j-1})$$

then we have an $(n-1)^{th}$ order Markov model or n -gram model. An n -gram is a sequence e_{i-n+1}^j consisting of a context e_{i-n+1}^{j-1} and a single-event prediction e_i . Since the use of a global order bound imposes assumptions about the nature of the data, the selection of an appropriate n is an issue when designing and building n -gram models. If the order is too high, the model will overfit the training data and fail to capture enough statistical regularity; low order models, on the other hand, suffer from being too general and failing to represent enough of the structure present in the data. The appropriate order for any particular corpus of data can only be determined experimentally.

An n -gram parameter is the probability of the prediction occurring immediately after the context. The parameters are typically estimated on some corpus of example sequences. Of several different means of estimating n -gram parameters, the simplest is *maximum likelihood* (ML) estimation which estimates the parameters as:

$$p(e_i|e_{i-n+1}^{j-1}) = \frac{c(e_i|e_{i-n+1}^{j-1})}{\sum_{e \in \xi} c(e|e_{i-n+1}^{j-1})}$$

where $c(e_i|e_{i-n+1}^j)$ denotes the frequency count for n -gram e_{i-n+1}^j .

Due to data sparseness, problems arise when using fixed order ML models due to the occurrence of as yet unseen n -grams. In particular, if a novel n -gram context is encountered or a novel symbol occurs in an existing context (the zero-frequency problem – see Witten & Bell, 1991), the ML estimate will be zero. In these situations, the estimated probability of a novel n -gram will be too low and consequently the estimated probability of n -grams with non-zero counts will be too high. In addition, the information theoretic performance measures that we shall use (see §2.2) require that every symbol is predicted with non-zero probability.

In statistical language modelling, a set of techniques known collectively as *smoothing* are commonly used to

address these problems. The central idea of smoothing is to adjust the ML estimates in order to generate probabilities for as yet unencountered n -grams. This is typically achieved by combining the distributions generated by an h -gram model with some fixed *global order bound* h with distributions less sparsely estimated from lower order n -grams. Most existing smoothing techniques can be expressed using the framework described in Equation 1 (Kneser & Ney, 1995).

If an n -gram e_{i-n+1}^j occurs with a non-zero count then the estimate $\alpha(e_i|e_{i-n+1}^{j-1})$ is used; otherwise, we recursively *backoff* to a scaled version of the $(n-2)^{th}$ order distribution $p(e_i|e_{i-n+2}^{j-1})$ where the scaling factor $\gamma(e_i|e_{i-n+1}^{j-1})$ is chosen to ensure that each conditional distribution sums to one: $\sum_{e \in \xi} p(e|e_{i-n+1}^{j-1}) = 1$. Recursion is typically terminated with the zeroth order model or by taking a uniform distribution over ξ . Different smoothing algorithms vary in the methods used for computing $\alpha(e_i|e_{i-n+1}^{j-1})$ and $\gamma(e_i|e_{i-n+1}^{j-1})$.

An alternative to backoff smoothing is *interpolated smoothing* in which the probability of an n -gram is always estimated by recursively computing a weighted combination of the $(n-1)^{th}$ order distribution with the $(n-2)^{th}$ order distribution as described in Equation 2.

$$p(e_i|e_{i-n+1}^{j-1}) = \alpha(e_i|e_{i-n+1}^{j-1}) + \gamma(e_{i-n+1}^{j-1})p(e_i|e_{i-n+2}^{j-1}) \quad (2)$$

Detailed empirical comparisons of the performance of different smoothing techniques have been conducted on natural language corpora (Chen & Goodman, 1999; Martin et al., 1999). One of the results of this work is the finding that, in general, interpolated smoothing techniques outperform their backoff counterparts. Chen & Goodman (1999) found that this performance advantage is restricted, in large part, to n -grams with low counts and suggest that the improved performance of interpolated algorithms is due to the fact that low order distributions provide valuable frequency information about such n -grams.

2.2 Performance Metrics

It is common in the field of statistical language modelling to use information theoretic measures to evaluate statistical models of language. Given a discrete random variable X distributed over an alphabet ξ according to a probability distribution P such that the individual probabilities are independent and sum to one, the entropy $H(P)$ is defined as:

$$H(P) = - \sum_{e \in \xi} p(e) \log_2 p(e) \quad (3)$$

Shannon's fundamental coding theorem (Shannon, 1948) states that entropy provides a lower bound on the average number of binary bits per symbol required to encode an outcome of $\mathcal{X}$. The corresponding upper bound occurs in the case where each symbol in the alphabet has an equal probability of occurring: $\forall e \in \xi, p(e) = \frac{1}{|\xi|}$.

$$H_{\max}(\xi) = \log_2 |\xi| \quad (4)$$

Entropy has an alternative interpretation in terms of the degree of uncertainty that is involved in selecting a symbol from an alphabet: greater entropy implies greater uncertainty.

In practice, we rarely know the true probability distribution of the stochastic process and use a model to approximate the probabilities in Equation 3. *Cross entropy* is a quantity which represents the divergence between the entropy calculated from these estimated probabilities and the source entropy. Given a model which assigns a probability of $p_m(e_i^j)$ to a sequence of outcomes of $\mathcal{X}$, e_1^j , we can calculate the cross entropy $H_m(e_1^j)$ of model m with respect to event sequence e_1^j simply by having very large sequences of outcomes available. In particular, if we make some assumptions about the stochastic process which generated the sequence, the cross entropy $H_m(e_1^j)$ may be calculated as:¹

$$H_m(e_1^j) = -\frac{1}{j} \sum_{i=1}^j \log_2 p_m(e_i | e_1^{i-1}) \quad (5)$$

Since $H_m(e_1^j)$ provides an estimate of the number of binary bits required on average to encode a symbol in e_1^j in the most efficient manner and there exist techniques, such as arithmetic coding (Witten et al., 1987), which can produce near optimal codes, cross entropy provides a direct performance metric in the realm of data compression. However, cross entropy has a wider use in the evaluation of statistical models. Since it provides us with a measure of how uncertain a model is, on average, when predicting a sequence of events, it can be used to compare the performance of different models on some corpus of data. In statistical language modelling, cross entropy measures are commonly used: "For a number of natural language processing tasks, such as speech recognition, machine translation, handwriting recognition, stenotype transcription and spelling correction, language models for which the cross entropy is lower lead directly to better performance." (Brown et al., 1992, p. 39).

¹In particular, we assume that the process is *stationary* and *ergodic*. A stochastic process is stationary if the probability distribution governing the emission of symbols is stationary over time (i.e., independent of the position in the sequence) and ergodic if sufficiently long sequences of events generated by it can be used to make inferences about its typical behaviour.

2.3 The PPM Algorithm

2.3.1 Overview

Prediction by Partial Match (Cleary & Witten, 1984) is a data compression scheme the central component of which is an algorithm for performing backoff smoothing of n -gram distributions. Variants of the PPM scheme have set the standard in lossless data compression since its introduction (Bunton, 1997). We shall describe several of these variants in terms of Equations 1 and 2 where recursion is terminated with a model which predicts each event $e \in \xi$ with equal probability mass, $\frac{1}{|\xi|}$. This model is usually referred to as the *order - 1* model and allows for the prediction of events which have yet to be encountered.

2.3.2 The Zero-frequency Problem and Escaping

We shall now describe how the probability estimates $\alpha(e_i | e_{i-n+1}^{j-1})$ and $\gamma(e_i | e_{i-n+1}^{j-1})$ in Equations 1 and 2 are computed in PPM models. The problem is usually characterised by asking how we estimate $\gamma(e_i | e_{i-n+1}^{j-1})$ – the amount of probability mass to assign to events which are novel in the current context e_{i-n+1}^{j-1} . $\alpha(e_i | e_{i-n+1}^{j-1})$ is then set such that the distributions sum to one. As noted by Witten & Bell (1991), there is no sound theoretical basis for choosing these *escape probabilities* in the absence of *a priori* knowledge about the data being modelled. As a result, although several schemes exist, their relative performance on any particular task can only be determined experimentally. In the following discussion, $t(e_i^j)$ denotes the total number of symbol types that have occurred in context e_i^j .

Method B (Cleary & Witten, 1984) classifies a symbol occurring in a given context as novel unless it has already occurred *twice* in that context. This is achieved by subtracting one from all the counts and has the effect of filtering out anomalies. In addition, the escape count increases as more types are observed.

$$\gamma(e_i | e_{i-n+1}^{j-1}) = \frac{t(e_{i-n+1}^{j-1})}{\sum_{e \in \xi} c(e_{i-n+1}^{j-1})}$$

$$\alpha(e_i | e_{i-n+1}^{j-1}) = \frac{c(e_i | e_{i-n+1}^{j-1}) - 1}{\sum_{e \in \xi} c(e_{i-n+1}^{j-1})}$$

Method C (Moffat, 1990) retains from method B the effect that the escape count increases as more types are observed but symbols are predicted immediately.

$$\gamma(e_i | e_{i-n+1}^{j-1}) = \frac{t(e_{i-n+1}^{j-1})}{\sum_{e \in \xi} c(e_{i-n+1}^{j-1}) + t(e_{i-n+1}^{j-1})}$$

$$\alpha(e_i | e_{i-n+1}^{j-1}) = \frac{c(e_i | e_{i-n+1}^{j-1})}{\sum_{e \in \xi} c(e_{i-n+1}^{j-1}) + t(e_{i-n+1}^{j-1})}$$

One particular smoothing technique called *Witten-Bell smoothing*, often used in statistical language modelling, is based on escape method C (Manning & Schütze, 1999).

These escape methods have been subjected to empirical evaluation in data compression experiments which demonstrate that method C typically yields better performance than method B (Bunton, 1997; Moffat et al., 1994).

2.3.3 Interpolated Smoothing

We have discussed the difference between backoff and interpolated smoothing in §2.1 and shown how they can be described within the same framework. While the original PPM algorithm uses a backoff strategy (called *blending*), Bunton (1997) has experimented with using interpolated smoothing within PPM. Bunton notes that for blending (and other backoff methods), the estimates for novel events are slightly inflated while the estimates for events which are not novel are slightly deflated. Replacing blending with interpolated smoothing remedies this and yields significant and consistent improvements in compression performance (Bunton, 1997).

2.3.4 Unbounded Length Contexts

One of the goals of *universal* modelling is to make minimal assumptions about the nature of the stochastic processes responsible for generating observed data. As we discussed in §2.1, n -gram models make assumptions about these processes to the effect that the probability of an event depends only on the previous $n - 1$ events. Cleary & Teahan (1997) describe an extension to PPM, called PPM*, which eliminates the need to impose an arbitrary order bound. The policy used to select a maximum order context can be freely varied depending on the situation.

A context e_i^j is said to be *deterministic* when it makes exactly one prediction: $t(e_i^j) = 1$. Cleary & Teahan (1995) have found that for such contexts the observed frequency of novel events is much lower than expected based on a uniform prior distribution. As a consequence, the entropy of the distributions estimated in deterministic contexts will tend to be lower than in non-deterministic contexts. Since the event will have occurred at least as many times in the lowest order matching deterministic context as any of the other matching deterministic contexts, it will produce the lowest-entropy probability distribution (Bunton, 1997). Cleary & Teahan (1997) exploit this in PPM* by selecting the shortest deterministic matching context if one exists or otherwise selecting the longest matching context. Unfortunately, the original PPM* implementation provided modest improvement in compression performance over the original order bounded PPM. When combined with interpolated smoothing, however, PPM* does outperform the corresponding order bounded PPM models in data compression experiments (Bunton, 1997).

2.3.5 Implementation Issues

Since PPM* does not impose an order bound, all subsequences of the input sequence must be stored which makes for increased demands on computational resources. Suffix-tree representations provide a space-efficient means of achieving this end (Bunton, 1997; Larsson, 1996). We have implemented our PPM models as suffix trees using the online construction algorithm described by Ukkonen (1995). The application of this algorithm to the construction of PPM models was first described by Larsson (1996) and the construction developed independently by Bunton (1997) is similar to the Ukkonen-Larsson algorithm in many respects. In addition to being online, these algorithms have linear time and space complexity and, as demonstrated by Bunton (1997), the resulting models have optimal space requirements (in contrast to the original PPM* implementation). The existence of path compressed nodes in suffix trees complicates the storage of frequency counts and their use in prediction. We have followed the strategies for initialising and incrementing the counts employed by Bunton (1997) to address these complications.

2.4 Long- and Short-term Models

In data compression, a model which is typically empty initially is constructed incrementally as more of the input data is seen. However, experiments with PPM using an initial model that has been derived from a training text demonstrate that pre-training the model, both with related and with unrelated texts, significantly improves compression performance (Teahan & Cleary, 1996). A complementary approach is often used in the literature on statistical language modelling where improved performance is obtained by augmenting n -gram models derived from the entire training corpus with *cache* models which are constructed dynamically from a portion of the recently processed text (Kuhn & De Mori, 1990).

Conklin (1990) has employed similar ideas with music data by using both a *long-term model* (LTM) and a *short-term model* (STM). While the LTM parameters are estimated on the entire training corpus, the STM is constructed online for each composition in the test set and is discarded after the relevant composition has been processed. The predictions of both models are combined to provide an overall probability estimate for the current event. The motivation for doing so is to take advantage of recently occurring n -grams whose structure and statistics may be specific to the individual composition being predicted.

A simple way of achieving the combination of predictions from the LTM and STM is to use a weighted average of the individual predictions (Conklin, 1990). Let $e \in \xi$ be the current symbol to be predicted, M be a set $\{lrm, stm\}$ containing the LTM and STM and $p_m(e)$ be the probability assigned to symbol e by model $m \in M$. The weighted mean of the two predictions is:

$$p(e) = \frac{\sum_{m \in M} w_m p_m(e)}{\sum_{m \in M} w_m} \quad (6)$$

Conklin describes a method for calculating the weights, w_{lrm} and w_{stm} based on the entropies of the distributions generated by the LTM and STM such that greater entropy (and hence uncertainty) is associated with a lower weight. Let P_m be the probability distribution generated by model m . The *relative entropy* of a model is:

$$H_{relative}(m) = \begin{cases} H(P_m)/H_{max}(\xi) & \text{if } H_{max}(\xi) > 0 \\ 1 & \text{otherwise} \end{cases}$$

where H and H_{max} are as defined in Equations 3 and 4 respectively. The weight w_m of model m is computed as $w_m = H_{relative}(m)^{-b}$ where $b \in \mathbb{N}$ is a parameter giving an exponential bias towards models with lower relative entropy. The combined use of long- and short-term models yields better prediction performance than either the LTM or STM used individually (Conklin, 1990). Finally, Conklin & Witten (1995, p. 61) have used a different scheme based on the Dempster-Schaffer theory of evidence for combining the predictions of long- and short-term models "with some success" but do not provide any details of the scheme or the performance improvements it yielded.

3 Related Work

N -gram models have been used for music related tasks since the 1950s when they were investigated as tools for composition and analysis (see e.g., Brooks et al., 1957; Hiller & Isaacson, 1959; Pinkerton, 1956). Since extensive reviews of this early research exist (Ames, 1989; Hiller, 1970), we shall focus here on more recent approaches.

Ponsford et al. (1999), for example, have applied trigrams and tetragrams (without smoothing) to the modelling of harmonic movement in a corpus of 84 seventeenth century *sarabandes*. The aim was to find out how adequate a simple n -gram model would be for the description and generation of harmonic movement in the style. Higher order structure was represented in the corpus through the annotation of events delimiting bars, phrases and entire pieces. A number of pieces were generated from the models and subjected to an informal stylistic analysis. The generated harmonies were "characteristic of the training corpus in terms of harmony transitions, the way in which pieces, phrases and bars begin and end, modulation between keys and the relation between harmony change and metre" (Ponsford et al., 1999, p. 169). The generation of features such as enharmony, which was not present in the corpus, and weak final cadences was attributed mainly to the use of low order models.

Conklin & Witten (1995, see also Conklin 1990) developed PPM models of the soprano lines of 100 of the

chorales harmonised by J.S. Bach. The escape method used was B and both long- and short-term models were employed. The global order bounds of the LTM and STM were set at 3 and 2 respectively and the predictions combined using a Dempster-Schaffer scheme. One of the central features of this work was the representation of multiple attributes, or *viewpoints*, of a melodic sequence. Conklin & Witten (1995) describe a number of *multiple viewpoint systems* consisting of several PPM models trained on different viewpoints whose predictions were combined in the same manner as described in §2.4. Several evaluation techniques were employed. First, split-sample validation with a training set of 95 compositions and a test set of five compositions was used to compare the performance (in terms of cross entropy) of different multiple viewpoint systems. Conklin & Witten were able to derive multiple viewpoint systems whose entropy was significantly lower than that of a single viewpoint system modelling chromatic pitch. The second means of evaluation was a *generate-and-test* approach from which Conklin & Witten concluded that the generated compositions seemed to be "reasonable". Finally, Witten et al. (1994) conducted an empirical study of the sequential chromatic pitch predictions made by human listeners on the same test set of compositions. The entropy profiles derived from the experimental results for each composition were strikingly similar in form to those generated by the model.

Hall & Smith (1996) have extended the approach used by Conklin & Witten (1995) to a corpus of 58 twelve-bar blues compositions. The aim was to develop a compositional tool that would automatically generate a melody when supplied with a twelve-bar blues harmonic structure. In order to model pitch, zero, first and second order models were derived from 48 compositions in the corpus. Separate first and second order models were derived for each individual chord occurring in the corpus. Rhythm was represented using an alphabet of short rhythmic patterns (e.g., two semiquavers followed by a quaver) and zero, first and second order models were derived from the training set over this alphabet. The model was evaluated by asking 198 human subjects to judge which of a pair of compositions (of which one was human- and the other machine-composed) was machine-generated. The data consisted of the ten remaining compositions in the corpus and ten compositions randomly selected from the model's output all of which were played to the subjects over a standard harmonic background. Statistical analysis of the results demonstrated that the subjects were unable to distinguish reliably between the human and machine generated compositions.

Reis (1999) has extended the work of Conklin & Witten (1995) in a different direction through the incorporation of psychological constraints in n -gram models. In particular, he argues that storing all n -grams (with order less than the global bound) which occur in the data is highly inefficient and unlikely to accurately depict the manner in which humans represent melodies. Reis describes a

ID	Description
0	Folk songs and Ballads from Nova Scotia, Canada
1	Chorale soprano melodies harmonised by J.S. Bach
2	Alsatian folk songs from the Essen Folk Song Collection
3	Yugoslavian folk songs from the Essen Folk Song Collection
4	Swiss folk songs from the Essen Folk Song Collection
5	Austrian folk songs from the Essen Folk Song Collection
6	German folk songs from the Essen Folk Song Collection (<i>kinder dataset</i>)
7	Chinese folk songs from the Essen Folk Song Collection (<i>shanxi dataset</i>)

Table 1: Melodic datasets used in this research.

model which segments the data according to perceptual cues such as contour changes or unusually large pitch or duration intervals. The order of the n -grams stored by the model is then determined by the sequence of events back to the previous segmentation point. If a novel n -gram is encountered during prediction, the distribution delivered by the variable order model is smoothed with a uniform distribution over the alphabet. The model also incorporates perceptually guided predictions for more than one step ahead. The performance of the model was evaluated on the chorale dataset used by Conklin & Witten (1995) and folk melodies from the Essen Folk Song Collection (Schaffrath, 1995) using entropy as the performance metric with a split sample experimental design. Although the results demonstrated that the model failed to outperform that of Conklin & Witten (1995), the work is useful since it addresses the question of which segmentation and modelling strategies work best when model-size is limited.

4 Experimental Methodology

4.1 Model Parameters

A PPM model has been implemented in Common Lisp such that each of the variant features described in §2.3 are accepted as parameters to the top-level call. We shall use the following shorthand to refer to each of the model parameters: the model type is indicated by 'LTM' and 'STM' for the long- and short-term models respectively; the escape method is indicated explicitly by 'B' or 'C'; the order bound is indicated by an integer or '*' if unbounded; and interpolated smoothing is indicated by an 'I' (blending is the default). Thus, for example, a PPM long-term model with escape method C, unbounded order and interpolated smoothing is denoted by 'LTMC*I'. When combined with a short-term model with the same parameters, the model would be denoted by 'LTMC*I-STMC*I' (for readability the two models are separated by a hyphen).

4.2 Data

The aim of this research was to assess the performance of PPM variants over a range of different musical styles.

The datasets used were all obtained in the `**kern` format (Huron, 1997) from the *Centre for Computer Assisted Research in the Humanities* (CCARH) at Stanford University, California (see <http://www.ccarh.org>) and the *Music Cognition Laboratory* at Ohio State University (see <http://kern.humdrum.net>). During preprocessing, tied notes were collapsed together and the chromatic pitch of each event was converted into a MIDI note number where 60 represents middle C. Each composition therefore consists of a sequence of integers each of which represents a chromatic pitch.

The datasets themselves contain purely melodic music. The first is a collection of 152 folk songs and ballads from Nova Scotia, Canada collected between 1928 and 1932 by Helen Creighton. The dataset is freely available from the *Music Cognition Laboratory* at Ohio State University. The second dataset contains 185 of the chorale soprano melodies harmonised by J.S. Bach (BWV 253 to BWV 438) and is freely available from CCARH. The remaining datasets come from the Essen Folk Song Collection (EFSC – Schaffrath, 1992, 1994). The collection comprises 6,252 (mostly) European folk melodies collected and encoded under the supervision of Helmut Schaffrath at the University of Essen in Germany between 1982 and 1994. A dataset containing all the compositions in the collection is published and distributed by CCRAH (Schaffrath, 1995) and an additional dataset of 2580 Chinese folk melodies is available on request. The six datasets from the EFSC used in this research contained respectively 91 Alsatian folk melodies, 119 Yugoslavian folk melodies, 93 Swiss folk melodies, 104 Austrian folk melodies, 213 German folk melodies (from dataset *kinder*) and 237 Chinese folk melodies (from dataset *shanxi*).

Each dataset is assigned a natural ID as shown in Table 1 and will be referred to henceforth by this ID. More detailed information about each dataset, including the number of compositions and events contained in the dataset and the number of chromatic pitches from which the dataset is composed, can be found in Table 2.

ID	No. Compositions	No. Events	Mean Events/Composition	Alphabet Size
0	152	8553	56.270	25
1	185	9227	49.876	21
2	91	4496	49.407	32
3	119	2691	22.613	25
4	93	4586	49.312	34
5	104	5306	51.019	35
6	213	8393	39.403	27
7	237	11056	46.650	41

Table 2: Detailed information about the datasets used in this research.

4.3 Performance Evaluation

Many methods have been used to evaluate the performance of statistical models of music, some of which have been described in §3. We have followed the resampling approach using entropy as a performance metric for two reasons: first, entropy has an unambiguous interpretation in terms of model uncertainty on unseen data (see §2.2); and second, entropy bears a direct relationship with performance in compression and indirectly correlates with the performance of n -gram models on practical natural language tasks and is widely used in both these fields. These factors support its use in an application independent evaluation such as this.

Conklin & Witten (1995) used a *split-sample* (or *held-out*) experimental paradigm in which the data is divided randomly into two disjoint sets, a training set and a test set; the n -gram parameters are then estimated on the training set and the cross entropy of the test set given the resulting model is computed using Equation 5. Conklin & Witten used a training set of 95 melodies and a test set of 5 melodies. Although commonly used, split-sample validation suffers from two disadvantages: first, it reduces the amount of data available for both training and testing; and second, with small datasets it provides a biased estimate of the true entropy of the corpus. A simple way of addressing these limitations is to use *k-fold cross-validation* in which the data is divided into k subsets of approximately equal size. The model is trained k times each time leaving out a different subset to be used for testing and an average of the k cross entropy values thus obtained is then computed. In machine learning research, differences in model performance as assessed by resampling techniques, such as cross-validation, are often analysed for significance using statistical tests such as the t-test (Dietterich, 1998; Mitchell, 1997).

5 Results

To illustrate the performance improvements obtained with the PPM variants discussed in this paper, we have successively applied escape method C, unbounded orders and interpolated smoothing to our emulation of the model used by Conklin & Witten (1995) which is described in

our framework as LTMB3-STMB2 (see §2). The results are shown in Table 3 where each result was obtained using ten-fold cross-validation. Paired t-tests confirmed the significance of the improvements afforded by incrementally applying escape method C [$t = 32.53, df = 79, p < 0.001$], unbounded orders [$t = 7.34, df = 79, p < 0.001$] and interpolated smoothing [$t = 24.57, df = 79, p < 0.001$].²

6 Discussion

Before discussing the results presented in §5, some words on the methodology employed are in order. Our goal was to demonstrate that a number of techniques improve the prediction performance of PPM models on monophonic music data. We have approached this task by using cross entropy of the models as our performance metric and applying ten-fold cross validity resampling on eight monophonic datasets. Since we have been concerned with optimising average performance over all eight datasets, the best performing model will not necessarily correspond to the best performing model on any single dataset. However, the results increase our confidence that this model will perform well on a given dataset without requiring further empirical investigation of that dataset.

Furthermore, since we have applied the variant techniques incrementally, there is no guarantee that the resulting model reflects the global optimum in the space of possible LTM and STM parameterisations. Once again, we note that our aim was to demonstrate how some variant techniques can improve the performance of PPM models and consequently our interest is in the relative, rather than absolute, performance of the PPM variants.

In our experiments, escape method C yielded consistent improvements in model performance over method B. Although, as noted in §2.3.2, there is no principled a priori means of selecting the escape method, these results are consistent with those obtained in data compression experiments (Bunton, 1997; Moffat et al., 1994; Witten & Bell, 1991). The use of unbounded orders, as described in §2.3.4, improves the average performance of PPM models

²These tests were performed over all 10 resampling sets of each dataset ($n = 80$) which are not shown in Table 3.

Dataset	LTMB3-STMB2	LTM3-STMC2	LTM*-STMC*	LTM*-I-STMC*I
0	2.901	2.604	2.585	2.465
1	2.665	2.470	2.451	2.327
2	3.080	2.759	2.705	2.602
3	2.960	2.686	2.643	2.558
4	3.007	2.671	2.556	2.483
5	3.284	2.852	2.728	2.661
6	2.536	2.249	2.167	2.085
7	3.060	2.762	2.770	2.642
Average	2.937	2.632	2.576	2.478

Table 3: Performance improvements to our emulation of the model used by Conklin & Witten (1995).

though to a lesser extent and with less consistency across the datasets than applying escape method C. This finding agrees with those obtained in experiments on data compression (Bunton, 1997) and is likely to be due to the fact that the optimal order bound varies between datasets. As noted by Bunton (1997, p. 90), order bound experiments “provide more information about the nature of the test data, rather than the universality of the tested algorithms.” Finally, the use of interpolated smoothing consistently improves model performance regardless of the dataset. This is in agreement with results obtained in experiments in data compression (Bunton, 1997) and on natural language corpora (Chen & Goodman, 1999; Martin et al., 1999). The reason appears to derive from the fact that backoff smoothing (of which blending is an example) consistently underestimates the probabilities of non-novel events (Bunton, 1997) for which the low order distributions provide valuable frequency information.

7 Conclusions

By way of conclusion, we would like to present some directions that we feel would be profitable to explore in future research. The first set of suggestions concern model development. First, an empirical comparison of the performance of the Demster-Schaffer scheme used by Conklin & Witten (1995) to combine the predictions of the LTM and STM with other techniques (including the weighted mean employed here) would be useful for future model developers. Second, Bunton (1997) describes an information-theoretic state selection mechanism which replaces the original state selection used in PPM* (see §2.3.4) and which consistently improves performance in data compression experiments. It remains to be seen whether this mechanism can be fruitfully applied with music data. Finally, the extension of the methodology used in this research to comparisons between different modelling approaches could yield interesting results. It would be useful, for example, to compare the performance of the PPM variants analysed here with that of models using other smoothing techniques commonly used in statistical language modelling, such as Katz back-off (Katz, 1987) and Kneser-Ney smoothing (Kneser &

Ney, 1995), and models based on the Lempel-Ziv dictionary compression algorithm as used by Dubnov, Assayag and their colleagues (Assayag et al., 1999; Dubnov et al., 1998; Lartillot et al., 2001), the prediction suffix automata used by Lartillot et al. (2001) and Triviño-Rodríguez & Morales-Bueno (2001) and the neural network models described by Mozer (1994).

Our second set of suggestions concern the data used. It should be emphasised that we have restricted our attention to a single attribute of musical sequences: chromatic pitch. None of the conclusions reached in this research can be guaranteed to hold for other attribute domains and representations; we shall need similarly detailed experiments to assess whether the performance improvements recorded here remain valid with these new representations. Therefore, an important consideration is the extension of the approach to other attributes of musical events and more sophisticated representations of musical works. Conklin & Witten (1995), for example, describe several means of deriving more abstract representations of the musical surface as well as developing methods for combining the predictions of n -gram models of these representations. In a similar vein, we consider it important to extend the approach to homophonic and polyphonic music. The issue of representing such music for training statistical models is discussed by, for example, Assayag et al. (1999), Conklin (2002), Pickens et al. (2002) and Ponsford et al. (1999). Since the results obtained here are in broad agreement with those obtained in data compression and statistical language modelling experiments, we expect the performance improvements to hold some degree of generality and to carry over to these more sophisticated representations of music.

Our final suggestions are methodological. The first concerns the fact that many of the directions cited above concern comparisons between different models. Standard corpora exist for comparing model performance in both the data compression and statistical language modelling communities: e.g., the Calgary corpus (Bell et al., 1990) and LOB corpus (Johansson et al., 1986) respectively. Such standardisation facilitates the objective and empirical comparison of different models and would be highly beneficial to the music processing community. Another

methodological issue concerns the validity of entropy as a measure of performance; in order to address this question we need detailed empirical studies of the relationship between entropy measures and model performance on a range of musical tasks such as those outlined in §1. In the meantime, we hope that the techniques described in this paper can be profitably applied to practical musical tasks and that the consequent reduction in cross entropy will translate into actual performance improvement on these tasks.

Acknowledgements

We would like to thank Darrell Conklin and Kerry Robinson as well as two anonymous reviewers for their comments on previous drafts of this paper. Marcus Pearce is supported by EPSRC via studentship number 00303840.

References

- Ames, C. (1989). The Markov process as a compositional model: a survey and tutorial. *Leonardo*, 22(2), 175–187.
- Assayag, G., Dubnov, S., & Delerue, O. (1999). Guessing the composer's mind: applying universal prediction to musical style. In *Proceedings of the 1999 International Computer Music Conference*, (pp. 496–499). San Francisco: ICMA.
- Bell, T. C., Cleary, J. G., & Witten, I. H. (1990). *Text Compression*. Englewood Cliffs, NJ: Prentice Hall.
- Brooks, F. P., Hopkins, A. L., Neumann, P. G., & Wright, W. V. (1957). An experiment in musical composition. *IRE Transactions on Electronic Computers*, EC-6(1), 175–182.
- Brown, P. F., Della Pietra, S. A., Della Pietra, V. J., Lai, J. C., & Mercer, R. L. (1992). An estimate of an upper bound on the entropy of English. *Computational Linguistics*, 18(1), 32–40.
- Bunton, S. (1997). Semantically motivated improvements for PPM variants. *The Computer Journal*, 40(2/3), 76–93.
- Chen, S. F. & Goodman, J. (1999). An empirical study of smoothing techniques for language modelling. *Computer Speech and Language*, 13(4), 359–394.
- Cleary, J. G. & Teahan, W. J. (1995). Some experiments on the zero-frequency problem. In Storer, J. A. & Cohn, M. (Eds.), *Proceedings of the IEEE Data Compression Conference*. Washington, DC: IEEE Computer Society Press.
- Cleary, J. G. & Teahan, W. J. (1997). Unbounded length contexts for PPM. *The Computer Journal*, 40(2/3), 67–75.
- Cleary, J. G. & Witten, I. H. (1984). Data compression using adaptive coding and partial string matching. *IEEE Transactions on Communications*, 32(4), 396–402.
- Conklin, D. (1990). Prediction and entropy of music. Master's thesis, Department of Computer Science, University of Calgary. Available as Technical Report 1990–390–14.
- Conklin, D. (2002). Representation and discovery of vertical patterns in music. In Anagnostopoulou, C., Ferrand, M., & Smaill, A. (Eds.), *Proceedings of the Second International Conference of Music and Artificial Intelligence*, (pp. 32–42).
- Conklin, D. & Witten, I. H. (1995). Multiple viewpoint systems for music prediction. *Journal of New Music Research*, 24(1), 51–73.
- Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7), 1895–1924.
- Dietterich, T. G. & Michalski, R. S. (1986). Learning to predict sequences. In R. S. Michalski, J. Carbonell, & T. M. Mitchell (Eds.), *Machine Learning: an Artificial Intelligence Approach*, volume II (pp. 63–106). San Mateo, CA: Morgan Kaufman.
- Dubnov, S., Assayag, G., & El-Yaniv, R. (1998). Universal classification applied to musical sequences. In *Proceedings of the 1998 International Computer Music Conference*, (pp. 332–340). San Francisco: ICMA.
- Ferrand, M., Nelson, P., & Wiggins, G. (2002). A probabilistic model for melody segmentation. In *Electronic Proceedings of the 2nd International Conference on Music and Artificial Intelligence (ICMAI'2002)*, University of Edinburgh, Scotland.
- Hall, M. & Smith, L. (1996). A computer model of blues music and its evaluation. *Journal of the Acoustical Society of America*, 100(2), 1163–1167.
- Hiller, L. (1970). Music composed with computers – a historical survey. In H. B. Lincoln (Ed.), *The Computer and Music* chapter 4, (pp. 42–96). Cornell, USA: Cornell University Press.
- Hiller, L. & Isaacson, L. (1959). *Experimental Music*. New York: McGraw-Hill.
- Huron, D. (1997). *Humdrum and Kern*: selective feature encoding. In E. Selfridge-Field (Ed.), *Beyond MIDI: The Handbook of Musical Codes* (pp. 375–401). Cambridge, MA: MIT Press.

- Johansson, S., Atwell, E., Garside, R., & Leech, G. (1986). The tagged lob corpus. ICAME, The Norwegian Computing Centre for the Humanities, Bergen University, Norway.
- Katz, S. M. (1987). Estimation of probabilities from sparse data for the language model component of a speech recogniser. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(3), 400–401.
- Kneser, R. & Ney, H. (1995). Improved backing-off for m-gram language modelling. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 1, (pp. 181–184), Detroit, MI.
- Kuhn, R. & De Mori, R. (1990). A cache-based natural language model for speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(6), 570–583.
- Larsson, N. J. (1996). Extended application of suffix trees to data compression. In Storer, J. A. & Cohn, M. (Eds.), *Proceedings of the IEEE Data Compression Conference*, (pp. 190–199). Washington, DC: IEEE Computer Society Press.
- Lartillot, O., Dubnov, S., Assayag, G., & Bejerano, G. (2001). Automatic modelling of musical style. In *Proceedings of the 2001 International Computer Music Conference*, (pp. 447–454). San Francisco: ICMA.
- Manning, C. D. & Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press.
- Martin, S., Hamacher, C., Liermann, J., Wessel, F., & Ney, H. (1999). Assessment of smoothing methods and complex stochastic language modeling. In *Proceedings of the Sixth European Conference on Speech Communication and Technology*, (pp. 1939–1942), Budapest, Hungary.
- Mitchell, T. M. (1997). *Machine Learning*. New York: McGraw Hill.
- Moffat, A. (1990). Implementing the PPM data compression scheme. *IEEE Transactions on Communications*, 38(11), 1917–1921.
- Moffat, A., Sharman, N., Witten, I. H., & Bell, T. C. (1994). An empirical evaluation of coding methods for multi-symbol alphabets. *Information Processing & Management*, 30(6), 791–804.
- Moser, M. C. (1994). Neural network music composition by prediction: Exploring the benefits of psychoacoustic constraints and multi-scale processing. *Connection Science*, 6(2–3), 247–280.
- Pickens, J., Bello, J. P., Crawford, T., Dovey, M., Monti, G., & Sandler, M. B. (2002). Polyphonic score retrieval using polyphonic audio queries: A harmonic modelling approach. In *Proceedings of the third International Symposium on Information Retrieval*, (pp. 140–149). IRCAM: Paris, France.
- Pinkerton, R. C. (1956). Information theory and melody. *Scientific American*, 194(2), 77–86.
- Ponsford, D., Wiggins, G. A., & Mellish, C. (1999). Statistical learning of harmonic movement. *Journal of New Music Research*, 28(2), 150–177.
- Reis, B. Y. (1999). Simulating music learning: on-line, perceptually guided pattern induction of context models for multiple-horizon prediction of melodies. In *Proceedings of the AISB'99 Symposium on Musical Creativity*, (pp. 58–63). Brighton, UK: SSAISB.
- Schaffrath, H. (1992). The ESAC databases and MAP-PET software. *Computing in Musicology*, 8, 66.
- Schaffrath, H. (1994). The ESAC electronic songbooks. *Computing in Musicology*, 9, 78.
- Schaffrath, H. (1995). The Essen folksong collection. In D. Huron (Ed.), *Database containing 6,255 folksong transcriptions in the Kern format and a 34-page research guide*. Menlo Park, CA: CCARH.
- Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27(3), 379–423 and 623–656.
- Teahan, W. J. & Cleary, J. G. (1996). The entropy of English using PPM-based models. In Storer, J. A. & Cohn, M. (Eds.), *Proceedings of the IEEE Data Compression Conference*. Washington, DC: IEEE Computer Society Press.
- Triviño-Rodríguez, J. L. & Morales-Bueno, R. (2001). Using multi-attribute prediction suffix graphs to predict and generate music. *Computer Music Journal*, 25(3), 62–79.
- Ukkonen, E. (1995). On-line construction of suffix trees. *Algorithmica*, 14(3), 249–260.
- Witten, I. H. & Bell, T. C. (1991). The zero-frequency problem: estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory*, 37(4), 1085–1094.
- Witten, I. H., Manzara, L. C., & Conklin, D. (1994). Comparing human and computational models of music prediction. *Computer Music Journal*, 18(1), 70–80.
- Witten, I. H., Neal, R. M., & Cleary, J. G. (1987). Arithmetic coding for data compression. *Communications of the ACM*, 30(6), 520–541.

Computational Creativity by Structural Analogy between Acoustic Bayesian Models

Michael A. Casey

*Centre for Computational Creativity
Department of Computing
City University, London
casey@soi.city.ac.uk

Abstract

We present a method for generative modeling of audio content that performs mappings between minimum entropy hidden Markov models learnt from audio data. By training with a minimum entropy prior, compact, low-complexity models of the latent structure in audio source samples are obtained. Synthesis of new audio content is achieved by mapping the state sequence of a nominated *structure* model onto a nominated *content* model. The mapping is chosen such that the cross-entropies between the state variables of the nominated models are minimised. This creates an analogy between the models' structures, even when the specific content of the models varies significantly. The re-mapped content state sequences are inverted to yield spectral vectors that consist of the higher-order pattern information of the structure model and the low-order spectral features of the content model. To illustrate the methods, we present examples of mapping audio structure and content between drum beat samples in different styles.

1 Introduction

Audio content repurposing has a significant presence in musical culture. Much of the electronic music currently being produced is based on transforming sampled materials in some way, and there exist a number of commonly-used signal processing approaches for such transformations. Notable amongst them is the *phase vocoder*, (Portnoff, 1976), which uses the short-time Fourier transform for analysis and modified resynthesis of spectral data and has been used by composers for creating new acoustic content by spectral blending of two or more source sounds. Phase vocoding is a remarkably effective tool that enables a degree of independence between temporal and spectral transformations not possible with looping and re-sampling techniques. However, as with most signal processing transformations, it is agnostic to any patterns that exist in the audio data, thus the transformations are structurally naive. In this paper, we describe a structured cross-synthesis technique that maps higher-order structural features between audio samples; with structure determined automatically using minimum entropy learning methods for Bayesian models.

Hidden Markov models are widely used for acoustic modeling in speech and for classification of audio, (Rabiner, 1989) (Casey, 2002). This is due, in large part, to the Baum-Welch algorithm for maximum likelihood inference of model parameters. However, little work has been undertaken on using HMMs to synthesise audio content outside of the speech synthesis literature, (Yoshimura and Tokuday, 1998). We use HMMs to extract latent

structure from audio signals and use this structure for generating novel audio content that exhibits a blend of salient features from two source samples.

2 Structure Discovery HMMs

A large number of algorithms, and heuristics, exist for fitting models to data. But one must normally specify the structure of the model; for example, the number of states to use and the linkage of the transition matrix for hidden Markov models. To automatically learn such structure from features, we use a minimum entropy prior on the form of the internal variables for an HMM. This strategy combines the problem of model structure estimation with the problem of optimal parameter estimation from data and is solved using maximum *a-posteriori* (MAP) estimation. A hidden Markov model consists of multinomial parameters representing an initial state distribution, $\pi_i = P(s_1 = i)$ and between-state transition probabilities, $T_{ij} = P(s_{t+1} = j | s_t = i)$, for states $i, j \in \{1, \dots, K\}$ and time $t \in \{1, \dots, T\}$. States are parameterised by multidimensional Gaussian probability distributions over the space of observations. To compute the entropy of an HMM, the model parameters are concatenated into a stochastic vector $\theta = [\theta_1, \dots, \theta_N]$, thus creating a multinomial over the model's parameter space. The conditional *posterior* distribution, with respect to the model parameters θ and observable evidence $\mathbf{x}$, can be factored according to Bayes' rule:

$$P(\theta | \mathbf{x}) \propto P(\mathbf{x} | \theta) P_e(\theta) \quad (1)$$

with entropic prior $P_e(\theta)$ of the form:

$$P_e(\theta) \propto e^{-H(\theta)} = \exp \left[\sum_{i=1}^N \theta_i \log \theta_i \right] \quad (2)$$

Brand (1999) provides a method for maximum *a-posteriori* (MAP) estimation using the entropic prior, in Equation 2, that yields models that are biased towards representing the structure, sparsity and determinism inherent in the training data. Figure 1 shows a minimum entropy HMM of a 16-bar Latin percussion sample. Only the first two dimensions of the probability space are shown.

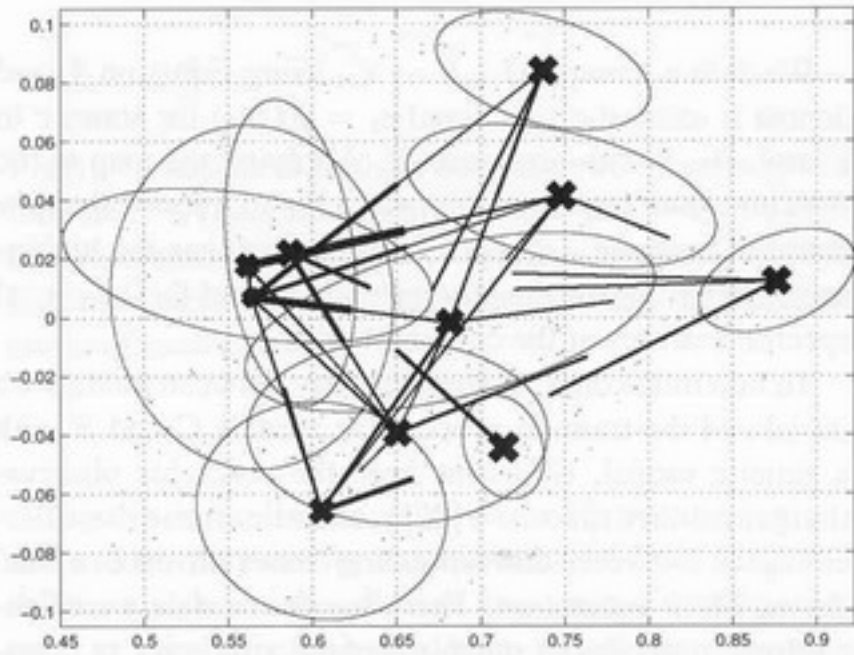


Figure 1: A 10-state hidden Markov model with sparse transition structure. Elipses represent Gaussian probability isocontours and lines represent transition probabilities between states. The size of 'X' in each state denotes the probability of self-transition.

Figure 2 illustrates the structure that is represented by the HMM state variables. Noisy low-level spectral audio features are converted into a representation that encodes the higher-order event structure in the data. It is easy to 'pick out' the states that correspond to onset and sustain phases of the percussive timbres in the figure.

2.1 Audio Feature Extraction

Log spectral features were extracted according to the specifications of the MPEG-7 low-level audio descriptors standard, see ISO (2001). Below, we also extract linear basis functions and low-dimensional projection coefficients using the MPEG-7 specification. Thus the feature-extraction and acoustic Markov modeling methods presented herein constitute an application of the MPEG-7 standard to structured content repurposing. We briefly describe these feature extraction methods in this section.

We recorded a set of 20 audio samples from a high-quality drum machine. Each sample consisted of a unique drum pattern in one of two styles (Latin or Techno) with a tempo of 75bpm and 16 bars duration in $\frac{4}{4}$ metre. Each sample was therefore 1280 spectral frames in duration and

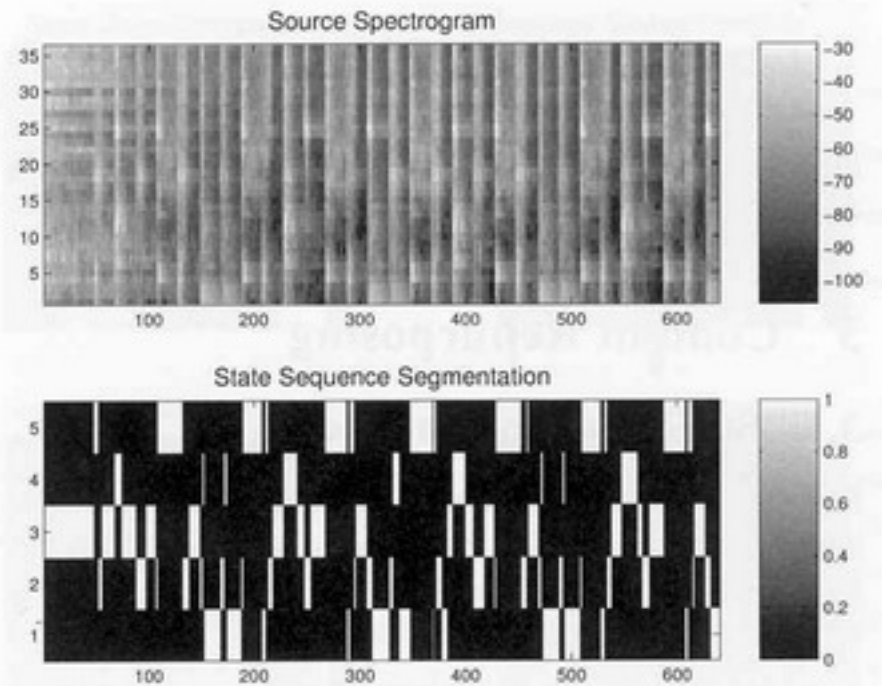


Figure 2: (Top) A constant-Q spectrogram of a Latin percussive sample lasting 6.4 seconds (10ms frames). (Bottom) Segmentation of the sample into 5 discrete states by a content-specific hidden Markov model.

the ensemble of training samples consisted of a total of 256 seconds of audio.

The signals were reduced to 16kHz mono 16-bit linear PCM samples and segmented by 30ms hamming windows, with a 10ms advance per frame, and transformed to the spectral domain using a 512-point FFT. The FFT elements were then converted to power spectrum coefficients and re-binned into $\frac{1}{8}$ th-octave bands, taking care to preserve the total power. The resulting log-spectral frames were 39-dimensional and preserved the power of the signal over the analysis window. These frames were stacked row-wise into observation matrices, $\mathbf{X}$, with the ensemble of stacked matrices representing all source samples, $\{\mathbf{X}\}$. Vectors were re-scaled to decibels and normalised by their L2-norm coefficients to yield unit-length spectral-shape vectors. The L2-norm coefficients were pre-pended to the ensemble vectors to create 40-dimensional vectors consisting of the RMS of spectral power coefficients in the first element and normalised spectral shape vectors in the remaining elements.

2.2 Audio Probability Space

To cross-synthesise audio data from multiple HMMs, one must ensure that the parameters of each model are defined over a common probability space. To do this, we first extracted short-time spectral features from each source audio sample, then we computed a linear basis over the spectral space.

The spectral linear basis was extracted by computing the singular value decomposition (SVD) over the ensemble of observation matrices, $\{\mathbf{X}\}$. The first five basis functions, $\mathbf{V}$, were retained and used to project observations into a low-dimensionality probability space:

$$\mathbf{Y} = \mathbf{XV} \quad (3)$$

This approach to audio feature extraction is described in detail, along with quantitative test experiments, in Casey (2002) and ISO (2001).

3 Content Repurposing

3.1 Structure, Content and Style

Methods for mapping between Bayesian models for style and content repurposing have previously been proposed in the fields of machine vision and computer graphics, (Brand, 2000) (Freeman and Tenenbaum, 1997). Brand's method uses learning to capture structure and style in bipedal motion performed by human subjects. Style hidden Markov models are used to synthesize new motion sequences by interpolating parameters between sample-specific models thus creating novel animations from existing content. Freeman and Tenenbaum (1997) use bi-linear models to separate style and content in images thereby enabling independent control over stylistic aspects of the data, such as the effects of lighting and pose in images of faces. There has also been some work in the area of speech synthesis on controlling HMM state variables for expressive articulation, (Yoshimura and Tokuda, 1998).

We adopted a similar framework for music and general audio repurposing. As such, this represents a departure from usual audio signal processing practices in that higher order structure is considered along with the low-order spectral structure.

3.2 Structure Mapping

Since the models were constrained to exist in a compatible probability space, by projection onto the linear basis V , the state parameters in different models could, meaningfully, be computed. Starting with two content-specific models trained on audio samples, we nominate one as the structure model, S , and the other as the content model, C . The best structural analogy was selected to be the map that minimised the sum of cross-entropies between all pairs of between-model states:

$$M(i) = \arg \min_j H_X(p_i; q_j) \quad (4)$$

The cross entropy, $H_X(p; q)$, between probability distributions $p(x)$ and $q(x)$ over $\mathbf{X}$ is defined as:

$$\begin{aligned} H_X(p; q) &= H(X) + D(p||q) \\ &= - \sum_{x \in X} p(x) \log(q(x)) \end{aligned}$$

with entropy $H(\mathbf{X})$ and relative entropy $D(p||q)$. Given two Gaussian states, p_i in model S and q_j in model

C , with mean and covariance parameters $(\mathbf{m}_1, \mathbf{K}_1)$ and $(\mathbf{m}_2, \mathbf{K}_2)$ respectively, and dimensionality d , the cross entropy calculated by:

$$\begin{aligned} H_X(p_i; q_j) &= \frac{1}{2} [d \log 2\pi e + \log |\mathbf{K}_i|] \\ &+ \frac{1}{2} [\log |\mathbf{K}_i| - \log |\mathbf{K}_j|] \\ &+ \sum_{mn} (\mathbf{K}_i^{-1})_{mn} (\mathbf{K}_j^{-1})_{mn} \\ &+ (\mathbf{m}_i - \mathbf{m}_j)^T \mathbf{K}_i^{-1} (\mathbf{m}_i - \mathbf{m}_j) - d] \end{aligned}$$

We define a map, $M : S \mapsto C$, using Equation 4, and denote it using the functional $c_t = M(s_t)$ for states c in C and s in S with time index t . Applying the map to the structure state sequence, $(s_1, s_2, \dots, s_T)$, returns the new content sequence, $(c_1^*, c_2^*, \dots, c_T^*)$, which has the higher-order patterning of the structure model but the low-level spectral features of the content model.

To maximize the structural analogy between models we initialized the training process for models C and S with a generic model, G , learnt from the ensemble observation probability space, $P(\{\mathbf{X}\})$, and minimized the cross-entropies between corresponding states in each model during MAP estimation. Therefore the models were constrained to exhibit as much structural similarity as possible whilst capturing the specific content of the individual samples from which they were trained.

Figure 3 illustrates structural analogy using the minimum cross-entropy map between two models of drum beat patterns. The top model was trained on a *Latin* style sample and the bottom was trained on *Techno* style sample. Using such maps we computed new audio sequences, by mapping the states of one model onto the states of the other.

3.3 Model Inversion

To invert the new state sequence, $(c_1^*, c_2^*, \dots, c_T^*)$, we used the means of the Gaussian states in C . Whilst this is clearly a simplification, constraining the emitted observations to visit only the centre of each state, it is sufficient for generating an approximation to the inverse of an HMM given a state sequence. New sequences of spectral acoustic vectors were generated by first projecting the mean vector of each state onto the inverse linear basis, V^T :

$$\mathbf{x}^* = \mathbf{m}_c V^T \quad (5)$$

then taking the first RMS power element from $\mathbf{x}^*$, multiplying it with the remaining elements, and inverse decibel scaling. The resulting log-frequency power spectrum coefficients were used to drive a filterbank for synthesis of an acoustic waveform.

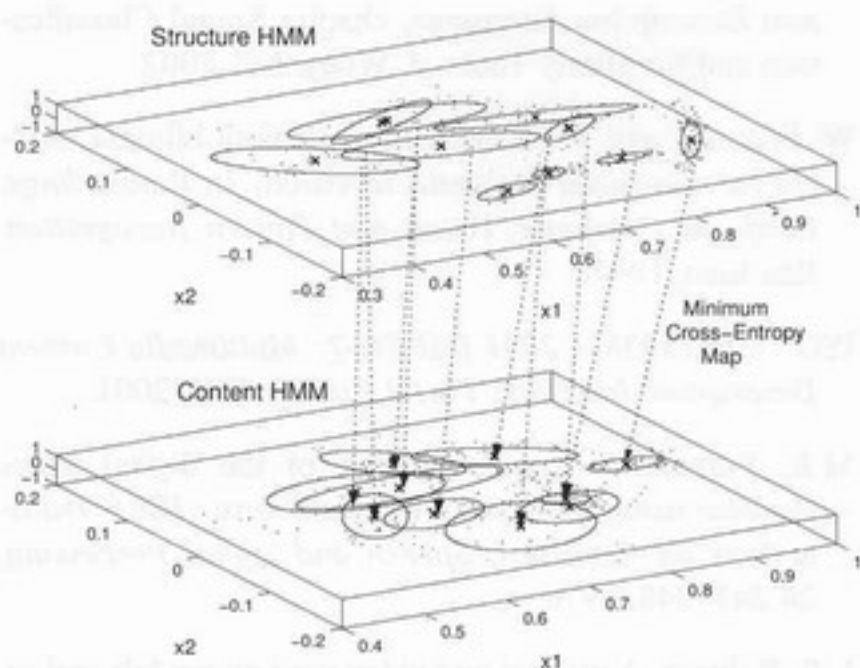


Figure 3: Structural analogy between two sample-specific models of a *Latin* drum sample (top) and a *Techno* drum sample (bottom). The arrows between-models depict the minimum cross-entropy mapping between the Gaussian states of each model.

Figures 4, 5 and 6 show the results of structure mapping and model inversion for combinations of drum patterns in different styles. In each case, two source samples were used to generate new hybrid state sequences. The entire process was automatic so the results reflect the ability of structure discovery methods to find salient higher-order structure in acoustic data without supervision.

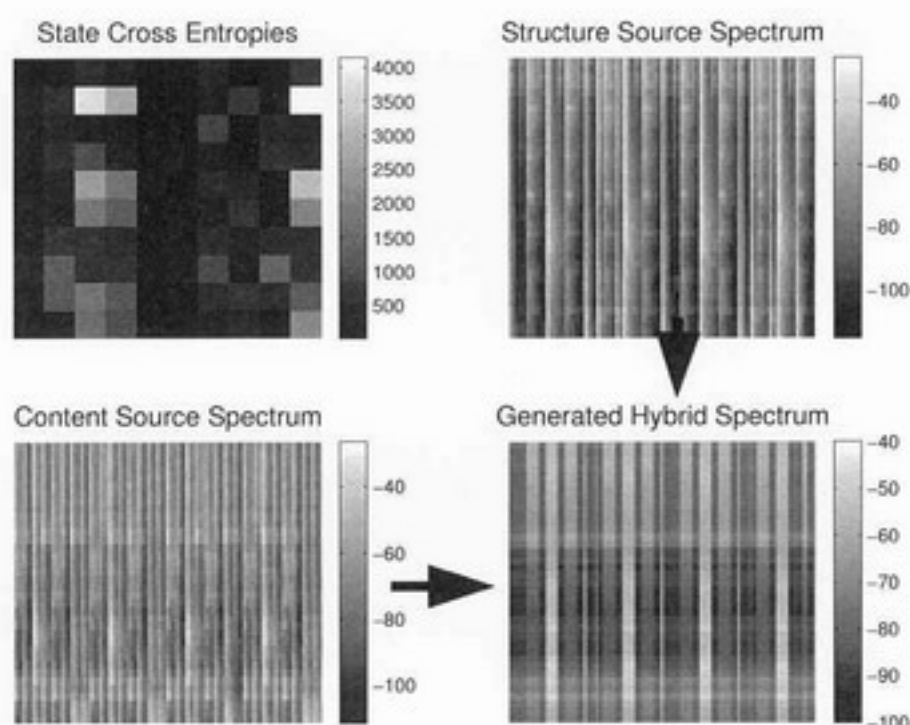


Figure 4: Generation of new structured audio content by remapping structure between source models (Top L) Cross entropy map of between-model states (Top R) spectrum of structure source [Latin:1] (Bottom L) spectrum of content source [Techno:1] (Bottom R) new spectrum.

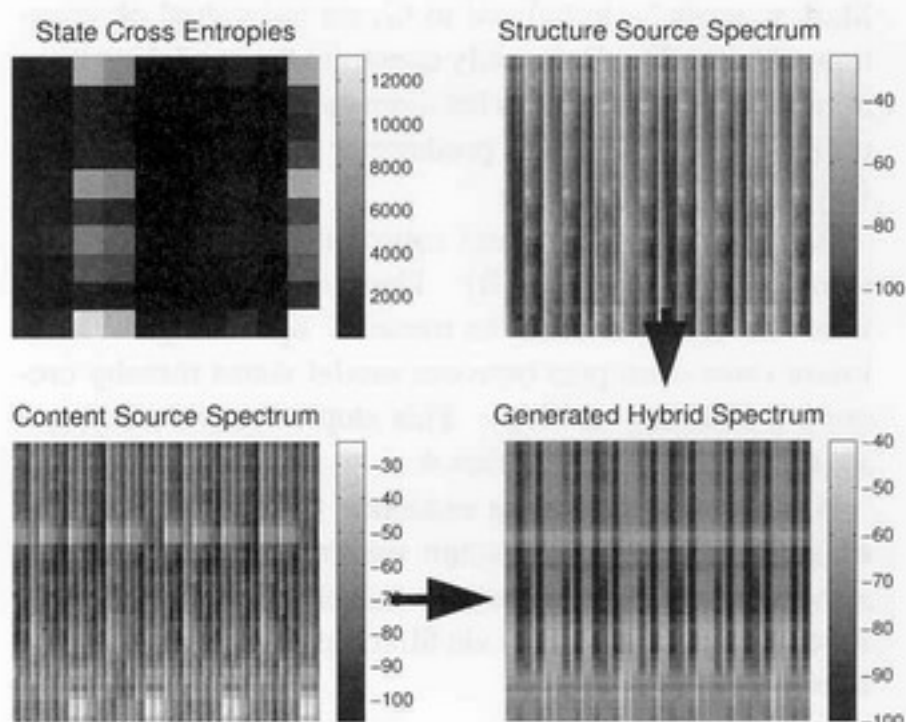


Figure 5: (Top L) Cross entropy map of between-model states (Top R) spectrum of structure source [Latin:2] (Bottom L) spectrum of content source [Techno:1] (Bottom R) new spectrum.

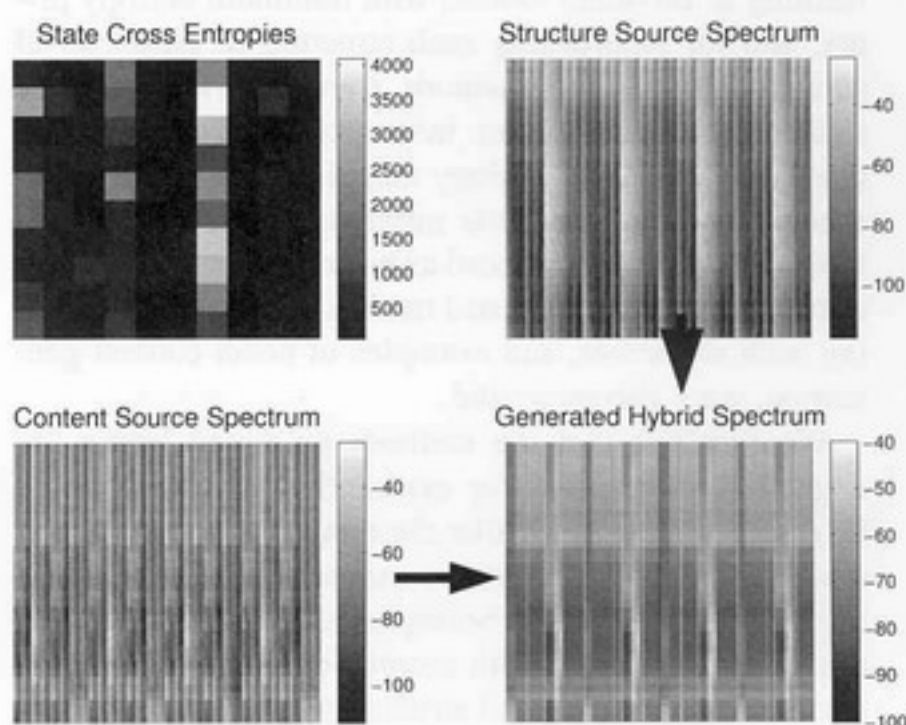


Figure 6: (Top L) Cross entropy map of between-model states (Top R) spectrum of structure source [Latin:1] (Bottom L) spectrum of content source [Techno:2] (Bottom R) new spectrum.

4 Summary

To summarise the method of content repurposing by structural analogy we enumerate the steps for training, mapping and inverting sample models.

1. Extract audio features from an ensemble of source samples, generating observation matrices $\{X\}$.
2. Extract an ensemble linear basis, V , by singular value decomposition of $\{X\}$.
3. Train a hidden Markov model on the ensemble observation matrix to produce a generic model G .
4. Train an ensemble of sample-specific hidden

Markov models, initialised to G , on individual observation matrices X_j . Optionally constrain the models to have minimum cross-entropies between model states after each training epoch. This step produces a set of source models $\{S\}$.

5. Nominate a structure source model and a content source model, $S, C \in \{S\}$. Then, map the state trajectories for model S onto the model C by finding the minimum cross-entropies between model states thereby creating a structural analogy. This step creates a new state sequence in C , see Equation 4.

6. Invert the new state sequence with respect to C to obtain a series of observation vectors and invert the observation vectors to generate a series of spectral frames. Invert the spectral frames via filterbank resynthesis to produce an audio signal.

5 Conclusion

In this paper we presented new techniques for automatically discovering structure in audio sample data using learning in Bayesian models with minimum entropy priors, and for repurposing such structure to create novel audio content. The methods were shown to capture salient event and pattern information in complex audio data. Structural analogy mapping between content-specific models, using the minimum cross-entropy between states, was introduced as means for generating new state sequences from trained models. Methods for inverting such sequences, and examples of novel content generation, were also presented.

We conclude that the methods presented herein are a good starting point for exploration of creativity by Bayesian inference. Whilst the methods produced good results for representing macrostructure, the current model inversion methods could be improved for generating spectral vector sequences with morphologies that better represent the micro-temporal structures in the training data. One such extension is to use a Bayesian method that maps smooth state trajectories in a structure model onto smooth state trajectories in a content model by learning the dynamic between-model relationships from pairs of training examples. Due to the additional complexity this has not yet been implemented, but will be the subject of future extensions to our methods.

References

M. Brand. Structure learning in conditional probability models via an entropic prior and parameter extinction. *Neural Computation*, 11(5):1155–1182, 1999.

M. Brand. Style machines. In *Proceedings of SIGGRAPH*, New Orleans, Louisiana, USA, 2000.

M. Casey. *Introduction to MPEG-7: Multimedia Con-*

tent Description Language, chapter Sound Classification and Similarity Tools. J. Wiley, NY, 2002.

W. Freeman and J. Tenenbaum. Learning bilinear models for two-factor problems in vision. In *Proceedings, Conf. on Computer Vision and Pattern Recognition*, San Juan, 1997.

ISO. *ISO 15938-4:2001 (MPEG-7: Multimedia Content Description Interface, Part 4:Audio)*. ISO, 2001.

M.R. Portnoff. Implementation of the digital phase vocoder using the fast fourier transform. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 24:243–248, 1976.

L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

T. Yoshimuray and K. Tokuday. Duration modeling for hmm-based speech synthesis. In *Proceedings of the International Conference on Speech and Language Processing*, Australia, 1998.