Postgraduate Conference on Computing: Application and Theory 2011

Proceedings of the 2nd Postgraduate Conference for Computing: Applications and Theory (PCCAT 2011)

8th June 2011
University of Exeter,
United Kingdom

Maximillian Dupenois and
David Walker (Eds.)

UNIVERSITY OF EXETER

Proceedings of the 2nd Postgraduate Conference for Computing: Applications and Theory (PCCAT 2011)

8th June 2011, University of Exeter, United Kingdom

Maximillian Dupenois and David Walker (Eds.)

Published by the College of Engineering, Mathematics and Physical Sciences, University of Exeter

Proceedings of the 2nd Postgraduate Conference for Computing: Applications and Theory (PCCAT 2011)

This event is kindly sponsored by

The works contained in these Proceedings of the 2nd Postgraduate Conference for Computing: Applications and Theory (PCCAT 2011) are reproduced by permission of the following contributors and owners of the copyright in the works:

Published by the College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter.

ISBN 978-0-9565982-1-9

Preface

PCCAT 2011 (Postgraduate Conference on Computing: Applications and Theory) is the second of the conferences in this series. Following on from the highly successful 2010 event, PCCAT 2011 has continued to showcase the high standard of work produced by postgraduate researchers in the south west of the United Kingdom. The focus of the conference is purposefully general so that researchers are able to receive feedback from a multi-disciplinary audience, possibly providing new insights into the problems being faced. This year we have been fortunate enough to receive work from fields as diverse as HCI (Human Computer Interaction) and multi-objective optimisation. This publication contains the papers presented at the second Postgraduate Conference on Computing: Applications and Theory (PCCAT 2011), held at the University of Exeter on June 8th, 2011.

The papers presented here all underwent review based on the intellectual rigour, novelty and style they displayed. Each paper was reviewed by at least two different reviewers and ample opportunity was provided for corrections. The conference series aims for quality within all its submissions, but approaches the reviews with the understanding that the papers are presented by postgraduates and a chance to improve is better than an immediate rejection. This process has led to the high quality of the papers published within these proceedings.

The day began with a keynote by Professor Steve Furber, the ICL Professor of Computer Engineering at the University of Manchester, which provided an interesting and thought-provoking start to the day. The papers were split into three sessions each session having a broad cohesive theme. These themes were Modelling and Design, HCI, and Evolutionary Computing. As well as the presented papers some postgraduates chose to display posters of their work during the lunch break. This break also provided a chance for people to network and socialise. The afternoon was split into two parts by a panel discussion featuring members from both the academic and industrial communities to consider the "Future of Computing" as it applies to early-career researchers, those who would be at the forefront of research into the new technologies and approaches. The day concluded with prizes being awarded for the best paper and the best poster. The prizes were kindly funded by the Society for the Study of Artificial Intelligence and Simulation of Behaviour (AISB).

We would like to thank both the Steering and Programme Committees for their hard work and support without which PCCAT 2011 would not have been possible. In addition we are grateful to the reviewers for the promptness of their responses and the dedication they showed in ensuring that the quality of presented work has been kept high. We are also thankful to the AISB for providing a substantial sum for the prizes. Finally we would like to acknowledge the sponsors of the conference, the University of Exeter and the University of Plymouth, the generosity, both fiscally and with their time, given by these institutions and their staff has been of immeasurable importance for the success of the day.

Organisation

Programme Chairs

Maximillian Dupenois	University of Exeter
David Walker	University of Exeter

Programme Committee

Samantha Adams	University of Plymouth
Maryam Astaraie-Imani	University of Exeter
Kent McClymont	University of Exeter
Joe Townsend	University of Exeter

Steering Committee

Kent McClymont	University of Exeter
Zena Wood	University of Exeter

Reviewers

Samantha Adams	University of Plymouth
Jacqueline Christmas	University of Exeter
Andrew Clark	University of Exeter
Christopher Ford	University of Plymouth
Kent McClymont	University of Exeter
Marek Rucinski	University of Plymouth
Benjamin Sanders	University of Plymouth
Hataichanok Saevanee	University of Plymouth
Joe Townsend	University of Exeter
Zena Wood	University of Exeter

Contents

Enhancing Voice Interaction by Providing Visual Feedback for an Assistive Robot

John Paul Vargheese

University of the West of England, Bristol

Abstract

The aging population of the UK has created over recent years a continued expansion and demand for care services for older people. Through assistive technology, older people can remain in their own homes for longer, reducing their cost of living compared to leaving their homes and moving into residential care. This research is a much smaller component (completed as part of a MSc dissertation) of a much larger EU funded research programme – MOBISERV (An Intelligent Home Environment for the Provision of Health Nutrition and Mobility Services to the Elderly) that seeks to develop a prototype assistive robot as part of a system to support independent living (www.MOBISERV.eu). In order for the MOBISERV platform to achieve its aims, it is important to explore how older people will interact with the system and the best means to achieve this. The robotic platform currently being developed provides speech recognition and synthesis capability, which is the primary means for older people to interact with the system, relay requests and receive feedback. Voice only interaction can often result in a high cognitive load for the user. The aim of this research is to investigate to what extent the efficacy of speech-based interaction can be improved by providing visual feedback. This involves evaluating different modes of visual feedback with a view to reducing cognitive load and enhancing user experience. As part of an iterative design and development process, usability studies with and without visual feedback help to discover how the interaction between the user and robot can be improved upon. Both qualitative and quantitative measures are being used to consider interaction in relation to a specific context, which is the creation of a shopping list. These findings will then be applied to other functions within the MOBISERV system implementation as part of future work.

1 Introduction

In a report published by OFCOM in March 2010, "Assisted living technologies for older and disabled people in 2030" Lewin et al. (2010) state that within five years robots could be used "to perform basic household tasks" that will provide both older people and people with disabilities to remain independent within their own homes for longer [1]. Ensuring efficient communication and interaction between the user and the system will be one of the key factors to ensure successful use and acceptance. Research into different methods of interaction is therefore a vital aspect of the development of these technologies.

1.1 The MOBISERV project

The EU funded MOBISERV project aims to provide the necessary means and support for older people to remain in their homes and retain a degree of independence through assistive technology. The MOBISERV project aims to provide a "personal intelligent platform" which supports independent living for older people (Welcome to MOBISERV, 2010) [2].

In order for the MOBISERV platform, and other assistive systems like it, to achieve its aims, it is important to consider how older people will interact with the system and the best means to achieve this. The main interactive element of the MOBISERV system is a robot which provides speech recognition capability. This is the primary means for older people to interact with the system, relay requests and receive feedback.

1.2 Advantages and disadvantages of voice interaction

One of the main advantages of voice interaction is the fact that speech is a natural, unobtrusive mode of communication or signal that is easily obtained from users, particularly those who will often be familiar with using speech for applications such as the telephone [2].

Voice interaction can also provide additional advantages for ease of managing email, streamlining repetitive computer based tasks and increasing the speed of information turnaround [3]. The obvious advantages of such benefits include reducing the time a user would spend preparing a message for communication applications such as instant messaging or email as well as completing a task using voice commands, rather than navigating through various menus or manually typing in commands. In addition to these advantages, speech recognition can also reduces the risk of musculoskeletal disorders (MSDs) and repetitive stress injuries (RSI) as the user is free from the keyboard and maintaining a single position or posture while using a computer [3].

Jackson (2005) provides a practical example of a speech recognition based application that has been included as a part of a robot's primary functions. This is a medical diagnostic system which asks the user or patient questions about discomfort or pain which the user maybe suffering from and uses the user or patients response as means to determine the cause and what action should be taken [4].

Additional benefits of voice interaction are for specific user groups with various impairments such as those with visual impairments and those who may suffer from motor-impairments [6].

In consideration of the various advantages and benefits that voice interaction provides, it is clear that these could enhance the level of interaction between older people and the system via the robot, as speech is a natural mode of communication and therefore provides both ease and flexibility. From reducing the time spent preparing an email, letter or reply to a received instant message to more critical application such as the diagnostic tool, speech recognition offers a wide range of potential benefits for enhancing and improving interaction between older people and the robot, to access the system's functionality.

However, various research studies have highlighted the flaws in applications which are based on voice interaction. These criticisms can be broadly separated into technical considerations related to configuration and performance issues, to more psychological and sociological considerations related to the overall interaction between the user and the machine.

In regards to technical considerations, specifically related to configuration, include problems related to voice training (as each user has a different voice to the next), problems related to interference caused by background noise in the user's environment, the need for manual correction of words, especially newly introduced words, phrases and errors caused as a result of noise interrupts, changes in voice (e.g. as a result of a cold), training time for the machine to learn a user's voice and retraining for new words [6]. Ceaparu et al. (2004) highlight the consequences of such issues are an increase in user frustration with the system, the results suggest that such problems, and consequent user frustration with the system, can result in approximately 50% of the users time wasted attempting to use a computer [7]. These performance and technical issues not only reduce the overall effectiveness of the speech recognition based applications, but can also result in lower acceptability and user satisfaction, particularly if the user experience and interaction with the system, frustrates the user and results with tasks incomplete and user goals not achieved.

Shneiderman (2000) argues that the application of speech recognition for effective user interfaces is flawed on the basis that human-human relationships, where speech is both effective and appropriate, is not an adequate model for human-computer-interaction [6]. Speech makes it difficult for the user to edit or review their input and that the information provided by the user is relayed, presented and entered slowly [6].

Gardner-Bonneau and Blanchard (2008) provide a categorised list of potential problems which may arise with existing speech recognition based systems and issues to consider upon designing them. These include problems of performance in regards to contemporary speech recognition systems, such as correcting errors, confined or restricted vocabulary and grammar related issues that may occur with the system either misinterpreting input and or generating incorrect output as a result of any combination of these problems [8]. Gardner-Bonneau and Blanchard (2008) also highlight problems and limitations of voice user interfaces due to the consecutive and gradual nature of working memory capacity and low constancy [8].

Furthermore Gardner-Bonneau and Blanchard (2008) also point out that additional problems may arise as a result of the difficulties arising from attempting to provide for both expert and novice users, as well as additional considerations that need to be made in relation to hardware (microphones, speakers, selected channels) and support provided by the computing platform or operating system [8].

1.3 Voice interaction design issues

1.3.1 GUI to VUI Design

Yankelovich et al. argue that based on the findings of their research, speech-only interfaces should not be designed as a translation from graphical alternative [13]. Specific problems the authors highlight in regards to transferring GUI designs to VUIs (Voice User Interfaces) are issues arising in the difference between the vocabularies used for GUIs in comparison to the vocabulary used for work or task based conversation. An example provided from the results revealed that when a participant attempted to use words and phrases such as

"next Monday" and "a week from tomorrow." The GUI based system has no concept of the meanings behind these phrases as they are not required, however these are phrases almost certainly used in conversation in the same context.

Furthermore Yankelovich et al. [13] argue that the presentation and information organisation of GUIs also does not transfer satisfactorily to the VUI. Participants were confused when an attempt to number emails as with the GUI based version of the system (which had been successful), failed to improve the usability and effectiveness of the application. The authors also highlight similar problems in relation to information flow which again confused participants when attempting to follow GUI design concepts such pop up dialogue boxes. Despite an attempt to create a speech based dialogue box, users often ignored the dialog prompts or became confused despite the system only requiring a limited response such as yes, no, ok etc. Yankelovich et al state that this behaviour is natural and provide an example as follows: If you are asked what the time is, the listener has to derive from the response a yes or no reply, usually from a larger dialogue. Some of the participants who replied to such prompts acted similarly, adding yes or not to their intended next action. This has severe implications for applications which rely on yes or no replies in order to progress with a particular course of action necessary to complete a user requested task [13].

1.3.2 Multimodal Interaction

A study conducted in order to develop a prototype conversational based system implemented on the Ford Model U Concept vehicle included a touch screen and speech recognizer for functions such as climate, entertainment, navigation and telephone. The authors raise issues with similar vehicle interfaces in regards to the need to press a button or select a control and then speak a command [14]. This is known as the command and control paradigm (Heisterkamp 2001) [15]. Pieraccini et al. highlight how these commands are not natural and are based on an artificial language developed which are difficult for the user to learn [14]. This paper discusses two major improvements to this system, the first being the implementation of a conversational based interface in order to reduce the cognitive load of the user and the second to provide support to the system with the aid of a GUI with a haptic touch screen. The GUI provides indicators and feedback on the dialogue state and recommendations on what to say as well as feedback on received input [14].

The main concept behind the design of the system was based on the requirement for the user to read documentation and instructions for use of the system, therefore ensuring that the system was truly intuitive. Catering for expert users by making the system flexible to user preferences (customisation) and adopting the principle of "do not speak until spoken to" was also a major design concept to be incorporated in order to make the system less intrusive [14].

1.3.3 Efficacy of Multimodal Interaction

Cohen et al. conducted a study which compared a "direct manipulation-based GUI" with a "QuickSet pen/voice multimodal interface for supporting the task of a military force 'laydown'" [16]. The authors were seeking to study whether or not VUIs will be more efficient in comparison to other interfaces. Results gathered from evaluating a multimodal speech and GUI based system compared to a direct manipulation input application resulted in significant performance increases. Despite a low speech recognition rate of 68%, the results suggested a greater efficiency and performance increase in comparison to a direct keyboard based input interface for the same task.

The focus of this study is to determine whether the efficacy of voice user interaction can be improved by providing visual feedback

2 Overview of the VUI Application

The shopping list VUI allows the user to prepare a shopping list using speech. The user states which items and amounts they wish to add to a shopping list and the list, item and quantity are displayed. The system prompts the user for confirmation using speech before the items are added to the shopping list and the display of the shopping list is updated to show the contents. As well adding and removing items, the user can also edit existing item quantities and confirmation prompts are included in order to double check the system's interpretation of the user's commands. The user can also request to review the contents of the list, whereby the system will read the contents.

2.1 Requirements Specification

Requirements for the VUI were gathered by reviewing three online supermarket websites. Specific aspects of the interface were considered in the context of MOBISERV and were selected as priority requirements that must be implemented in the prototype. The following tables summarises both functional and non-functional requirements.

2.1.1 Functional Requirements

Requirement	Description
Shopping List Type Selection	The user must be able to select a shopping list type from a range of options or be able to define a new type. This could be in the form of an already prepared list including items which the user can expect to already be present therefore removing the need to browse for these items. The "type" is specified by the user.
Add Command	In order to create the list the user must be able to add items to the shopping list. This function has been implemented in several areas with online super market websites' interfaces, so that there are multiple means to add an item to the shopping list
Remove Command	Just as important as the need to add items to the shopping list, the interface must provide a means to remove items once they have been added. These functions are usually contained within the online shopping trolley or list on various super market websites.
Edit Command	Related to both add and remove commands, users must be able to amend, edit or adjust the values entered into the shopping list. This could be due to user error or simply the desire to change an item's quantity
Clear Shopping List	The facility to remove all items from the shopping list is also required so that if necessary, users can empty the contents and start a new list or exit the application knowing that no items will be processed
Review Command	Without having to scroll down a list manually using the mouse or keyboard, the interface must be capable of reading the lists current contents as well as offering the option to complete the shopping list, make an amendment, remove an item or continue shopping.
Speech Synthesis	The system must be capable of speaking to the user as this is the primary mode of interaction between the user and the system for completing the shopping list task.
Speech Recognition	The system must be capable of listening to and recognising the users replies and input in order to create the shopping list and navigate through the process of completing the shopping list.

2.1.2 Non-Functional Requirements

Requirement	Description
Self-Descriptive informative language	The VUI must keep the user informed as to which stage of the procedure they are currently engaged with. This is essential in order to orientate the user and ensure they understand what input is required from them in order to proceed and use the program
Support and guidance dialogue	Related to the requirement above, it is necessary to support the user's interaction throughout the procedure of preparing their shopping list, by instructing the user through dialogue as to what input is required from them at each stage of the process.
Error prevention	Attempting to prevent user errors for each stage of the process will improve usability. Including confirmation dialogue prompts for each stage of the process which allows the user to double check their chosen input as well as the systems interpretation of what they have selected.
Error recovery	While attempting to prevent errors is advisable, it is still highly probably that errors will occur. Therefore providing a means for the user to correct and recover from errors again improves the user experience
User determined pace of dialogue	The user should feel comfortable and at ease while using the system. Therefore the system should follow the pace of the user and await their responses rather than prompting for a response within a specified time period. Smith (2004) argues that users' can easily become frustrated if too much information is presented at once too quickly [26]. This can be prevented by designing the system to proceed at the pace of the user
Low conceptual complexity	The system should be easily learnable by the user and in order to ensure this, the VUI must be designed in a manner that presents the required concepts necessary for preparing a shopping list, simply and effectively.
Low procedural complexity	Reducing the complexity of the required sub tasks and overall procedure for preparing a shopping list should be as simple as possible in order to reduce the potential for user errors and uncertainty

2.2 Implementing Requirements

The requirements specification detail essential features of the system which were implemented into a Java Swing application which supported create, read, update and delete (CRUD) functionality. Speech synthesis and recognition was implemented using the Quadmore Java to Microsoft speech application interface bridge for Windows [27]. The Quadmore bridge uses a dynamic link library or DLL file which allows Quadmore speech synthesis and recognition java based methods to access the Microsoft SAPI [27]. Access to the bridge and application logic is controlled from a single class which instantiates the VUI display class and another class for handling user and system dialogue.

These classes run together in order to prompt the user and recognise their replies. String values are assigned for what the system is to say to the user and also for user replies (calling Quadmore speech synthesis or recognition methods [27]). Application logic is controlled by a series of conditional statements that determine what the user hears and interprets what they say in response in order to prepare their shopping list.

3 Design Methodology

3.1 Iterative Design Process

An iterative design process was chosen as the development strategy for the prototype VUI. This approach was chosen in order to reveal potential usability and functionality issues. Once these issues have been identified, additional features to address problems caused as a result can be incorporated into updated versions of the prototype.

Phase 1 of the development process consisted of developer and limited user testing in order to discover what additional features could be incorporated into future re-designs of the prototype.

3.2 Application Procedure

The procedure for preparing a shopping list consists of the user specifying a type of list before adding items to it. Once the user has stated which items they would like to add to the list they are required to specify a quantity before an item is added. Confirmation prompts are included for list type, which can either be an existing or newly defined type. Items and quantity are also checked using confirmation prompts. Visual feedback includes displaying what type of list has been created, an item and its quantity both before and after confirmation has been received. The Windows speech recognition icon is also viewable which displays what the system interprets as a response as well as a graphical display indicating the volume of speech from the user.

3.3 Prototype VUI

Two initial prototypes were developed and evaluated as part of phase 1. The first prototype provided participants with limited visual feedback (displaying the items in the shopping list) from the application itself but also by displaying the windows speech recognition display. Early testing demonstrated speaking and listening difficulties due to poor synchronisation between the user and the system. Further observations revealed that in some cases it was difficult for the user to be fully aware of what stage of the procedure the system was currently at, as well as what available commands could be spoken.

To address these issues, visual feedback was included in the next prototype. This included text based system prompts for each speech synthesised request and what available commands could be spoken in response. This was an attempt to address speaking and listening difficulties observed upon developer testing with the initial prototype. It was also decided that each participant should have an individual speech profile in order to better improve recognition rates during the Phase 1 study.

3.4 Phase 1 Study Evaluation

In the initial pilot phase of the study three participants were selected for the user Phase 1 test evaluations of prototype two. For the main MOBISERV project, the evaluation studies will include a significantly larger group of participants. Participant A has a background in IT support, participant B is retired and has a background in health and social care and participant C who has a background in health and social care as well as providing care for older adults.

3.5 Evaluation Procedure

Each participant was required to create a new speech profile and complete a single speech tutorial before conducting testing with the VUI. Three tests were completed in total, one of which required the user to select a pre-defined list, add, remove and amend items on the shopping list. The second test required participants to create a new defined list and also to add, remove and amend items. The third test was left for the participants to decide. User testing was recorded using a screen capture program.

4 Results

[The following section is a summary of the results from the phase 1 study. The results have been categorised into themes based on emerging issues and observations made during the phase 1 user evaluations.

4.1 Ability to complete tasks and recover from errors

Despite making progress through the required tasks after multiple recognition errors, the system failed to recognise commands which it previously had accepted from participant A. This created more frustration and

anxiety during the testing which could be detected by the changing tone of voice and body language of the participant. At many points during the evaluation participant A changed how they were replying to the system, varying pace, clarity, pronunciation and tone. The participant also attempted to say multiple commands at once despite being informed by both the developer and the visual feedback to only speak single commands as also indicated by the instructions sheet. Participant A also stated that more options should be available for specifying brand, category of items and quantity. Participant A was unable in almost all cases to recover from recognition errors and after several attempts, ignored the errors and moved on to the next stage of the task.

During the first test participant B repeated and stated multiple commands at once. However participant B quickly learned how to recover from errors and throughout the remaining tests only spoke single commands when prompted by the system.

The participant was able to complete all tasks with only a few recognition errors which they were able to recover from quickly and easily. Participant C also only used single commands and never repeated commands for a single prompt and seemed to synchronise their speech with the VUI using the onscreen prompts.

4.2 Feedback from participants

Participant A made several comments about using the system related to usability. The participant felt that the instructions were not clear enough and that despite being informed by both the instructions and the developer to only speak when the available commands were shown, the participant felt that this was confusing and that it needed to be more resilient to errors.

Unlike participant A, participant B stated that both on-screen instructions and the instructions sheet were very informative and clear. The participant made several comments about the system being appropriate for older adults due to the low procedural complexity and natural model of thinking in regards to shopping, specifically in regards to selecting an item and then specifying a quantity. This was in direct contrast to comments made by participant A. When questioned further about this, participant B stated that the simplicity was apparent due to both onscreen prompts and the spoken prompts by the system and also stated that after making some mistakes, they became familiar with what was expected and required in order to complete the task. Upon discussing their ability to use the VUI afterwards, participant C stated that they felt they were not rushed to use the interface and this was one of the key factors that made the system easy to use. They also commented about the low procedural complexity and intuitiveness of the system to guide them through the process, help them recover from errors and inform them of what to say and when to say it.

4.2.1 Recognition Errors and User Frustration

Participant A's feedback was mainly negative and stated that the major flaw with the system was related to the poor recognition rate and requested to repeat the training and use another microphone in order to address this problem. Participant A experienced the highest rate of recognition errors and frustration and was unable to recover from recognition errors in most cases.

4.2.2 Low Procedural Complexity

Positive feedback from participant A included comments about the low procedural complexity and with minor changes to the interface and instructions, together with improved recognition rates would make the system suitable for its intended purpose.

Upon asked about recognition errors and error recovery participant B again stated that once they had made a few errors and adjusted how they spoke to the VUI, it was obvious and apparent what was required in order to progress through each task. Participant B attributed this to the low procedural complexity of the system and dialogue.

4.2.3 Speech Therapy

Overall participant B provided very positive feedback and stated that the VUI had great potential and could be further developed for post-stroke rehabilitation speech therapy.

4.2.4 Benefits for older people

Participant C stated that such a system would be ideal for older people, particularly those in residential care. Drawing from their experience in working in this area, Participant C stated that the ability for a resident to make their own shopping list and have it processed would give them a feeling of empowerment and control of their lives, because such tasks usually have to be completed by carers at the home. Participant C stated that a key feature would be the ability not to have to choose from a limited selection of goods but to be able to specify any item and include details regarding its brand and type during the initial prompt for an item to be added to the shopping list. Upon informing the participant that the items could be specific in terms of brand and type at the first prompt for adding an item to the shopping list, the participant revealed that they believed the selection of goods was limited and that not having a restriction would be very favourable amongst older adults in care homes. The participant stated that this was because older adults often feel reluctant to specify specific brands and types of goods on their shopping lists which they give to the carers to shop for them and that they would rather go without, than feel that they had burdened the carer with this task.

5 Conclusion

The Phase 1 study provided an opportunity to test the prototype VUI and determine whether it could be further developed using an iterative development process. After preliminary testing was completed, the first prototype was amended to include visual feedback in order to address the problems associated with when to listen, when to speak and what to say.

Observation of the participants completing tests during the Phase 1 study, and analysis of the recorded screen capture data, suggest a whole range of factors that must be taken into consideration upon determining what type of visual feedback enhances voice interaction.

For example system recognition errors may be related to synchronisation problems of when to speak, due to the speech recognition engine not detecting the whole command spoken by the user. Informing the user of when to speak and when to listen however may not entirely be effective if for example the user provides multiple commands at once or is unsure about what commands can be used as a reply. Error detection and informative dialogue that informs the user how to recover from specific errors may help address this issue and increase usability; however if recognition errors persist due to poor recognition rates as a result of user errors made during speech profile training, the user is unlikely to be able to recover from errors regardless of how informative the system dialogue is.

Qualitative measures also must be taken into account, such issues as whether the user changes how they speak during speech training compared to VUI testing. How the user speaks related to task performance (both system and user) related to body language, specifically hand gestures, facial expression and non-verbal expressions; which were clearly evident during the Phase 1 tests. Observing such user behaviour and associating the results with whether the user was able to successfully complete a task in relation to the number of errors caused, will be used as the basis for further iterations of the prototype to investigate these issues.

Establishing the cause and relationship between system and user errors is also an issue that will be investigated in order to determine the efficacy of voice interaction. For example, as well as recognition errors due to poor performance during speech training, too many commands or reply options for the user to remember or read onscreen, may create additional complexity for the user while completing a task. Therefore reducing the available commands that the user can say as a reply, may help to reduce procedural complexity and improve task completion success rates, usability and user satisfaction. Again, in order to research this issue the prototype will be modified in order to restrict available reply commands and the results of testing will be compared with other versions where only this aspect of the interaction has been modified.

Visual feedback, displaying the system prompts, may also increase conceptual complexity, as opposed to using a summary of the prompts. This again is related to what, how and when information is displayed on the screen in order to inform the user of which stage of the process they are currently at. Speaking and listening synchronisation is also relevant. For example, summarising what user hear, so that even without listening attentively, they can still understand what they are required to say compared to displaying exactly what has been spoken by the system. Which strategy is the most effective can be established by testing these features separately and comparing the number, type, and causes of errors. This data in relation to qualitative data based on observations made during training and testing can also be used to draw conclusions as to which strategy was more effective or whether the quantitative data (such as high recognition rate errors) is misleading due to other factors such as changing tone or mode of speech during the training and VUI testing. Whether this occurred as a result of experiencing recognition errors causing frustration and changes in body language again may alter the tone and mode of speech.

The results gathered from the Phase 1 study support some of the advantages of speech recognition Reynolds (2002) [2] and Revis (2005) [3] mention in regards to the comments made by participant C about how the shopping list VUI could benefit older people; related to the natural nature of speech and the reducing the potential of MDS and RSI that may occur when entering such information using a traditional desktop setup. However as Gardner-Bonneau (2008) [8] point out, issues related to additional problems that may arise as a result of the difficulties from attempting to provide for both expert and novice users as well as additional considerations that need to be made in relation to hardware are clearly evident from the performance and results gathered during participant A's testing. Results gathered from Ceaparu et al. study (2004) [8] in regards to user frustration resulting in technical difficulties causing recognition errors were also evident during the Phase 1 study. Observing performance during training, and providing further training in order to reduce recognition errors may improve voice interaction and increase the potential for task completion during future user testing. This, together with variations in visual feedback, will form the basis of future prototype VUIs and user testing. This study was just a pilot done as part of a MSc dissertation and will be followed up with validating the results with more participants. The larger MOBISERV project will include much more substantial and thorough evaluations and testing.

5.1 Acknowledgements

This work was developed within the MOBISERV project (www.MOBISERV.eu) that is partially funded by the European Commission under the Seventh Framework Programme (FP7-248434). The author wishes to thank the Commission as well as all project members for their support. The author would also like to acknowledge the help, guidance and support of Praminda Caleb-Solly.

6 Appendices

6.1 Appendix A

Limitation category	Definition	Issues and potential problems
Speech recognition	Limitations of current speech recognition technology	Errors, finite vocabulary, grammar acoustic model
Spoken language	Limitations arising from characteristics of spoken language	Spontaneous, public, natural turn taking protocol, anthropomorphism, limited expressive power
Environment	Disturbances from the user's environment	Noise, multiple people speaking, interruptions
Human cognition	Properties of the human cognitive system	Sequential and slow, working memory capacity, low persistence, competition with verbal processing
User	Differences between and preferences among users	Task knowledge, expert / novice, speech competence
Hardware	Properties of the hardware used to implement a speech user interface	Channel, microphones, computing platform

References

[1] Lewin D, Adshead S, Glennon B, Williamson B, Moore T, Damodaran L, Hansell P, 2010, Assisted living technologies for older and disabled people in 2030 – A final report to Ofcom, AEGIS / Plum / Sagentia

[2] Reynolds, D. A. An Overview of Automatic Speaker Recognition Technology. In Proc. International Conference on Acoustics, Speech, and Signal Processing in Orlando, FL, IEEE, pp. IV: pages 4072-4075, 13-17 May 2002

[3] Revis M, 2005, The Case for Speech Recognition, For the Record, http://www.fortherecordmag.com/archives/ftr_042505p20.shtml, [accessed 23/08/2010]

[4] Jackson M, 2005, Automatic Speech Recognition: Human Computer Interface for Kinyarwanda Language, Makere University

[5] Modec Instruments, anon, n.d. http://modecideas.com/faq93.html?newitems, [accessed 25/08/2010]

[6] Shneiderman B, 2000, The Limits of Speech Recognition: Understanding acoustic memory and appreciating prosody, http://hcil.cs.umd.edu/trs/2000-01/2000-01.html, [accessed 23/08/2010]

[7] Ceaparu I,Lazar J, Bessiere K, Robinson J, Shneiderman B, 2004, Determining causes and severity of end-user frustration, International Journal of Human-Computer Interaction, 17(3), 333-256, Lawrence Eribaum Associates, Inc

[8] Gardner-Bonneau D, Blanchard H.E, 2008, Human factors and voice interactive systems, P.2-18, Springer

[9] Reddy U, Reddy T, 2007, A Voice User Interface for an Activity-Aware Wearable Computing Platform, Umea University

[10] McTear M.F. Spoken dialogue Technology: Enabling the conversational user interface ACM Computer Surveys CSUR, New York NY USA Volume 34 pages 90-169 2002

[11] Tucker D, 2003, Voice user Interface Design – Purpose and Process, MSDN Library, Speech Technologies Technical Articles, http://msdn.microsoft.com/en-us/library/ms994650.aspx, [accessed 15/09/2010]

[12] Nielsen J, 2003, Voice Interfaces: Assessing the Potential, useit.com- Jakob Nielsens's Alertbox, http://www.useit.com/alertbox/20030127.html, [accessed 15/09/2010]

[13] Yankelovich N, Levow G-A, Marx M, 1995, Designing SpeechActs: Issues in speech user interfaces, Proceedings of the SIGCHI conference on Human factors in computing systems, Denver, Colorado, United States, pages 369 – 376, ACM Press/Addison-Wesley Publishing Co

[14] Pieraccini R, Dayanidhi K, Bloom J, Dahan J-G, Phillips M, Goodman B R, Prasad K V, 2009, A multimodal conversational interface for a concept vehicle , Commun. ACM 47, 1 (January 2004), pages 47-49

[15] Heisterkamp, P. 2001 Linguatronic – Product Level Speech System for Mercedes-BenzCars, Proc. of HLT 2001, Kaufmann, San Francisco

[16] Cohen PR, Johnston M, McGee D, Oviatt SL, Clow J, 2000, The Efficacy of Multimodal Interaction for a map-based task, Proceedings of the sixth conference on Applied natural language processing (ANLC '00). Association for Computational Linguistics, Stroudsburg, PA, USA, pages 331-338

[17] McGee D, Cohen P.R, Johnston M, Oviatt S, Pittman J, Smith I, Chen L, Clow J, , The Efficiency Of Multimodal Interaction: A Case Study Proceedings of the sixth conference on Applied natural language processing (ANLC '00). Association for Computational Linguistics, Stroudsburg, PA, USA, 331-338

[18] Demiris G, Oliver DP, Dickey G, Skubic M, 2008 Finding from a Participatory evaluation for older adults, Technology and Health Care Volume 16, Number 2, IOS Press pages 111-118

[19] Massimi M, Baecker RM, Wu M, 2007, Using participatory activities with seniors to critique, build and evaluare mobile phones, Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility (Assets '07). ACM, New York, NY

[20] Nielsen J, Ten Usability Heuristics, useit.com, http://www.useit.com/papers/heuristic/heuristic_list.html, [accessed 21/09/2010]

[21] Zazelenchuk T, 2006, Heuristic Evaluation and its alternatives, userfocus.co.uk, http://www.userfocus.co.uk/articles/heuristics.html, [accessed 21/09/2010

[22] Travis D, Usability Expert Reviews: Beyond Heuristic Evaluation, userfocus.co.uk, http://www.userfocus.co.uk/articles/expertreviews.html, [accessed 21/09/2010]

[23] Somerville J, Stuart LJ, Barlow N, 2006, Easy Grocery: 3D Visualization in e-Grocery, computer.org, Plymouth.ac.uk

[24] Freman MB, 2006, Assessing the usability of online grocery systems to improve customer satisfaction and uptake, PhD thesis, School of Information Technology & Computer Science, University of Wollongong University of Wollongong

[25] Salisbury H, 2010, Online supermarket taxonomy,foolproof.co.uk, http://www.foolproof.co.uk/tesco-supermarket-taxonomy/, [accessed 23/09/2010]

[26] Smith M, 2004, Voice User Interface Design – Tips and Techniques, MSDN Library Technical articles, http://msdn.microsoft.com/en-us/library/ms994651.aspx, [accessed 08/02/2011]

[27] Szoghy B, 2009, Quadmore Java to Microsoft SAPI bridge for Windows version 2.5, Quadmore Software, http://www.quadmore.com/JAVA_to_SAPI/, [accessed 04/03/2011]

[28] P. R. Cohen and S. L. Oviatt, "The Role of Voice Input for Human-Machine Communication," Proceedings of the National Academy of Sciences, vol. 92, 1995.

[29] J. P. Marshall, "A manufacturing application of voice recognition for assembly of aircraft wire harnesses," in Proceedings of Speech Tech/Voice Systems Worldwide. New York, 1992.

[30] G. L. Martin, "The utility of speech input in usercomputer interfaces," International Journal of Manmachine Studies, vol. 30, pp. 355-375, 1989.

[31] D. Visick, P. Johnson, and J. Long, "The use of simple speech recognisers in industrial applications," in Proceedings of INTERACT'84, First IFIP conference on Human-Computer Interaction. London: International Federation of Information Processing Societies, 1984.

[32] A. G. Hauptmann and A. I. Rudnicky, "A Comparison of speech and typed input," in Proceedings of the Speech and Natural Language Workshop. San Mateo, California, 1990, pp. 219-224.

[33] B. A. Mellor, C. Baber, and C. Tunley, "Evaluating Automatic Speech Recognition as a

Component of a Multi-Input Human-Computer Interface," in Proceedings of the International Conference on Spoken Language Processing, 1996.

Aquila: Massively Parallelised Developmental Robotics Framework

(Abstract)

Martin Peniak, Anthony Morse, Christopher Larcombe, Salomon Ramirez-Contla and Angelo Cangelosi

The University of Plymouth

The developmental robotic approach often relies on complex and computationally intensive algorithms running in parallel to control real-time systems. The processing requirements are increasing with every added feature and it is not uncommon that at the end of the software development stage a particular system is unable to cope with real-time robot-control tasks.

Recent technological advances in GPU (Graphics Processing Unit), e.g. the new NVIDIA architecture called CUDA (Compute Unified Device Architecture), now permits the programming of massively parallel GPU devices. These are in most cases significantly outperforming standard processors on parallel tasks (e.g. Jang et al. 2008, Jung, 2009).

Aquila is an open-source software application for developmental robotics inspired by the recent advancements in GPU parallel processing. This was developed as a part of the iTalk project (italkproject.org), on the integration of actions and language in humanoid robots. This software facilitates such integration and allows developers to share and reuse their work within one unifying framework was required but not present and therefore we started the Aquila project.

Aquila not only aims to provide an easy access to often used many modules, tools and applications but also aims to bring different researchers together to work on Aquila in a collaborative way, which will bring about the high-level integration of multiple subsystems. Aquila is primarily designed for use with the iCub humanoid robot or its simulator (Tikhanoff et al., 2008) but also provides several stand-alone tools that can be used by any one who does not have access to the iCub robot or its simulator such as self-organising maps, multiple timescale recurrent neural networks and training algorithms and others.

Aquila has been developed to be compliant with CUDA GPU processing. Benchmarking tests of Aquila showed significant speedups, in some cases over 1000x faster. These studies were applied to the MTRNN neural network control architecture.

Kolibri-A: a lightweight 32-bit OS for AMD platforms

Artem Jerdev

School of Physics, CEMPS, University of Exeter

Abstract

An attempt has been made to apply the KolibriOS operation system for use within fast technical vision systems. As a result, a fast and capable Kolibri-A system has been developed and specially optimized for modern PC platforms with AMD chipsets.

1 Introduction

The bandwidth of modern CMOS image sensors can exceed 1GB/s which can set very strict constraints on graphics efficiency, computational power, bus speed and OS response time. The latter is very significant for many general-purpose PC platforms equipped with powerful-but-slow "universal" operating systems.

The initial target of this work was to develop a software package to control a custom-built CMOS camera, acquire a broad-band video stream, show it on a screen, analyse the data in runtime and store the results on a hard drive. Attempts to build such a system based on a Linux platform failed due to both unacceptably slow system delays and the oversophisticated structure of the linux drivers involved [1].

After a lot of searching for a faster and simpler OS a short review article was found [2] which advised to try KolibriOS, one of the smallest and fastest graphical open-source operating systems for PC-platforms ever developed [3].

KolibriOS forked off the MenuetOS project in 2004 and was hugely optimized for productivity and size [4]. It has an extraordinary small kernel: version 0.7.7 takes up only 73kb of disk space and could reside inside a CPU cache.

The OS kernel is entirely written in the FASM assembly language [5] and supports a capable graphical user interface, pre-emptive multitasking, page translation, disk DMA, dynamic library linking, and many other features. User applications have a MeOS-like executable format with a set of unique system functions carefully optimized for speed, size, and resource management efficiency.

2 KolibriOS: tiny but powerful

First experiments with KolibriOS proved its outstanding speed and high stability. Most of the system calls take less than a thousand CPU clocks which explains the system's very short response time and makes it quite popular on slower platforms [6]. The kernel code contains many bright and effective programming solutions, being carefully optimized both on size and speed.

Table 1 compares the speed of some KolibriOS GUI elements tested with CPU AthlonIIx3(3GHz) on AMD RS780 platform. Corresponding WindowsXP (™) ratings shown in the right columns

Table 1

GUI speed, *elements per second*	KolibriOS	WinXP, VESA	WinXP, accelerated
Horiz. lines (300pix)	321,000	29,500	426,000
Random pixels	3.32 M	1.01M	167,000
Text lines (34 chars)	39,700	2,300	23,600

But despite its extraordinary speed and tiny size, KolibriOS still cannot be recommended for time-critical embedded applications. One of the principal obstacles is its primitive hardware subsystem which is totally dependent on slow and ineffective BIOS services.

There are two possible ways to improve this OS and enable use the full power of newer PC platforms. One way is to implement a full-featured device enumerator based on ACPI standards. Such an approach would guarantee a smooth system start and compatibility with most modern PC platforms, but will not operate equally effectively on every platform. It would also significantly increase the kernel size as well as system startup time.

Another idea is to optimize the system for *one selected platform*, and to control all of the hardware resources directly, ignoring any ACPI methods.

This method will:

- exploit PC resources with the best efficiency;
- minimize the system size;
- void kernel dependency on BIOS services;
- allow one to build a full-featured *ROM-resident* OS.

The obvious disadvantage of such an approach is its evident incompatibility with other platforms.

3 Kolibri-A

The new branch "Kolibri-A" has been developed to support fast-imaging experiments in the School of Physics, CEMPS, University of Exeter. It is an open-source product with a GPL-like licence.

The suffix "-A" shows the essential focus to AMD-specific features. Kolibri-A currently runs on modern AMD platforms with fam. 0F-10h processors and RS780+/SB700 chipsets [7, 8]. It can also run on the newest AMD Fusion platforms [9].

Besides the general KolibriOS features, it benefits from some unique solutions briefly listed below.

Table 2

	KolibriOS	Kolibri-A
User access to I/O resources	`ports`	`ports, MMIO`
User access to PCI config. space	`CF8/CFC`	`PCI Express extended cfg`
System service calls	`int40h`	`int40h, syscall`
Graphics supported	`EGA, VGA, VESA1, VESA2+, ATI r100+`	`VESA2, ATI r600, AMD Fusion`
Packed kernel size	`74 kb`	`63 kb`

3.1 User-accessible I/O resources

A user-space application now can request direct access both to I/O ports and memory-mapped I/O regions of the selected device. The system modifies the application's port map correspondingly and remaps the MMIO region to the application's linear address space. That gives the application a way to communicate with the device directly at the highest possible speed with no need of intermediate drivers.

3.2 PCI-e extended configuration space

Access to PCI configuration space is now using an extended memory-mapped mechanism to allow full-feature control of PCI Express devices [10].

User-space applications can also use this mechanism via special configspace-remapping functions. Such a unique feature allows users to develop direct (driverless) device control software and to trace configuration registers of non-standard PCIe devices and prototype boards. It also simplifies the process of fine hardware adjustment.

A set of advanced PCI Express Link & Flow control system utilities and demos has been included within the Kolibri-A repository [11]. These utilities proved to be a powerful tool of hardware diagnostics and let us localize some malfunctions of prototype PCI Express acquisition boards used.

3.3 Faster system calls

New AMD-specific syscall/sysret [12] system calls have been implemented in Kolibri-A. Such a mechanism takes only 90 CPU clocks to enter Ring0 and get back to the user space – three times faster than the standard interrupt-aided method which requires.

Both system calling conventions coexist in Kolibri-A to guarantee full compatibility with older software.

3.4 Optimized window manager

Most of reviewers appreciate compact and remarkably fast GUI of KolibriOS [4, 6]. But a less-known fact is that its minimalistic and resource-hungry graphics engine can seriously slow down other system services. For instance, its primitive window manager uses a huge block of cacheable memory (1 byte per each screen pixel) and can occasionally displace critical system code and data from CPU cache.

A new window manager has been developed for Kolibri-A. It stores the window stack information in a compact list and does not obstruct CPU cache.

Some GUI functions (i.e. font and bitmap graphics) have been greatly optimized for speed. The total size of graphics-related routines has been reduced by 30% and now takes less than 10kB of kernel code.

Table 3

GUI speed, *elements per second*	KolibriOS	Kolibri-A
Horiz. lines (300pix)	`321,000`	`399,000`
Vert. lines (350pix)	`23,500`	`23,600`
Proportional text (lines, 34 chars ea.)	`39,700`	`40,500`
Fixed-width text	`40,800`	`42,000`
Random pixels	`3,320,000`	`3,689,000`

3.5 Selected hardware control

KolibriOS kernel uses simple but ineffective VESA VBE mechanisms [13] to control graphics devices. Few device-specific drivers were ported from Linux. At the moment, only one driver (for ATI Radeon cards) can provide access to the full GPU power; other devices can only work in 2D VESA modes.

As it was said above, Kolibri-A was specially optimized for the AMD RS780 chipset [7] which includes an integrated ATI R600 graphics processor. That certainty allowed to replace legacy VESA modes and optimize the graphics driver for exclusive R600 support.

Evergreen GPU drivers have been recently added to support the newest AMD Fusion [9] APU graphics.

Also, the existing USB and audio drivers were greatly improved and simplified thanks to well-documented register model of AMD SouthBridges [8].

3.6 Diskless system boot

KolibriOS is known as the only 32-bit multi-task graphical OS that entirely fits to one 1.44M diskette [4]. Although many other variants of booting exist [14], the 1.44M disk image remains the main boot option so far.

To satisfy higher requirements of modern embedded systems, a packed version of Kolibri-A kernel has been stored to a PC ROM and booted disklessly in accordance with the BIOS boot specification [15]. This method takes less than 3 seconds from a cold start to all-ready state and can be very useful for diskless embedded systems and "sealed" specialized devices.

4 Conclusion and Future Work

The results of the new OS'es first tests show it can be successfully used for fast technical vision systems and time-critical data processing. It has been developed to capture and process fast broadband video streams and successfully tested with a custom-made CMOS camera (700Mpps).

Kolibri-A inherits the best features of its mother system and even beats the latter in speed and code size.

It also provides some unique instruments for time-critical data acquisition devices and hardware testing and development.

Such a combination of high speed, small size and powerful functionality might only be possible because all the low-level system codes were optimized for the newer AMD processors and chipsets with no care of compatibility with the other platforms.

Few things still need to be resolved to satisfy higher embeddable OS standards:

- new APIC-based interrupt management subsystem with a full-featured MSI support;
- new Task manager with a real-time process control;
- RT-optimized Memory manager;
- better support of SATA and USB buses.

Acknowledgements

I greatly appreciate KolibriOS Development Team [16] support and especially grateful to Evgeny Grechnikov and Sergey Semenov for their priceless advice.

References

[1] J.Corbet, A.Rubini, G.Kroah-Hartmann. Linux Device Drivers. *O'Reilly*, 2005.

[2] Alternative OSes: OS discovery pack. *Linux Format*, 117(4):72-73, 2009.

[3] www.kolibrios.org

[4] http://distrowatch.com/weekly.php?issue=20090831#feature

[5] http://flatassembler.net

[6] http://www.codigobit.info/2010/08/ejecutando-kolibrios-en-una-netbook.html

[7] AMD SB780 Register Reference Guide. *AMD P/N 43451* , 2008.

[8] AMD SB700/710/750 Register Reference Guide. *AMD P/N 430009* , 2009.

[9] BIOS and Kernel Developer's Guide for AMD Family 14h Models 00-0Fh Processors. *AMD P/N 43170* , 2011.

[10] A.H.Wilen, J.P.Shade, R. Thornburg. Introduction to PCI Express. *Intel Press*, 2003.

[11] http://redmine.kolibrios.org/projects/kolibrios/repository/show/kernel/branches/Kolibri-A

[12] AMD64 Architecture. Programmer's Manual Vol.1. *AMD P/N 24592 Rev.3.15*, 2009.

[13] VESA BIOS Extension. Core Functions Standard. *Video Electronics Standard Assotiation*, 1998.

[14] http://wiki.kolibrios.org/wiki/FAQ/

[15] BIOS Boot Specification. Ver.1.01. *Compaq-Phoenix-Intel*, 1996.

[16] http://wiki.kolibrios.org/wiki/Developers/

Using Digital Cultural Probes as a Requirements Elicitation Tool for System Design

Alison Flind, Praminda Caleb-Solly

University of the West of England

.

Abstract

Used as a compliment to other techniques, Cultural Probes enable researchers to gather an extremely rich data set. Traditional requirements gathering methods such as focus groups or interviews can often put people on the spot and may be constrained by issues such as time, or biased by the researchers' pre-conceptions. Cultural Probes elicit a different kind of data. Rich, multi-layered stories integrating users' real-world routines with their aspirations and perceptions of physical appearances within particular contexts. They give system designers a 'feel' for and deeper insight into the lives of their potential system users. Probes present clues about people's lives and thoughts, entwining emotional reactions with facts to inspire innovative technologies that can enrich the user experience.

The traditional Cultural Probe approach has been modified in recent years by a number of industrial and academic groups who are motivated by a desire to rationalise the data. Rather than appreciating the playful, subjective approach embodied by the original probes, some researchers have adapted them to ask specific questions and produce quantitative results.

During a pilot study modelling user requirements for a fashion retail website, *Digital* Cultural Probes were adopted as one of several methods for eliciting information. Users collected and submitted data in situ as they were experiencing events, using their mobile phones. This paper discusses this novel approach, giving a description and evaluation of the methods used, reiterating the benefits of implementing probes as they were originally intended to be used; as a tool for enhancing the data set.

1 Introduction

1.1 Cultural Probes

Cultural Probes [1] are a method for understanding users by applying collections of evocative tasks to elicit inspirational responses *from* people, rather than comprehensive information *about* them. Focussed on empathy and engagement, Probes offer fragmentary clues regarding users lives and thoughts that are gathered in context. Cultural Probes compliment other established requirements gathering techniques which are generally practised in less immersive environments, prompting traditional methodological concerns such as researcher bias or the Hawthorne effect. [2]

"Rather than producing lists of facts about our volunteers, the Probes encourage us to tell stories about them, much as we tell stories about the people we know in daily life... Over time, the stories that emerge from the Probes are rich and multilayered, integrating routines with aspirations, appearances with deeper truths. They give us a feel for people, mingling observable facts with emotional reactions.". [3, p. 5]

It is an approach that values uncertainty, play, exploration and subjective interpretation

Fig. 1. Traditional Cultural Probe Pack, Image source: http://www.hcibook.com/e3/casestudy/cultural-probes

As shown in Fig. 1., a Cultural Probe 'pack' might include disposable cameras, maps, booklets or postcards with instructions for users to complete.

2 Probe Evolution

Since they were first described by Gaver and his colleagues in 1999, probes have been adapted by researchers for many uses. In a comprehensive review by Boehner et al. [4] the use of probes has been categorised in relation to different objectives of research study. These include the use of Cultural Probes for rapid and broad data collection and as a tool for encouraging more user participation in the design process. They also identified the use of probes in a provocative manner to force new interactions and reactions by users.

2.1 Concerns

In 2004 Gaver expressed concern that Cultural Probes have been misappropriated and rationalised by users intent on *analysing* the data [3]. He reiterated that they should be used as an approach to encourage subjective engagement and that they should not be summarised or averaged out. Researchers should not ask unambiguous questions, analyse or justify the results. He suggests that empathetic interpretation and a pervasive sense of uncertainty should be embraced as positive values in the system design.

2.2 Applications

Gaver's original probe study was applied as part of a research project to investigate novel interaction techniques for older adults in their local communities. In this instance, lo-fidelity methods of data gathering such as stamped, addressed postcards (to be filled in and posted back to the researcher) or disposable cameras were appropriate as they are more likely to be used in these contexts as they are familiar to this user group and integrate into their everyday lives more naturally

2.3 Problems

These methods are not without their drawbacks - some researchers have reported low-return rates, with the probes disrupting users routines and being very time-consuming [5].

3 Digital Cultural Probes

3.1 Method

This research was based on an exploration of how people make decisions when purchasing consumer products. The aim of the investigation was to generate interesting insights to inform the development of requirements for designing online shopping applications. An online t-shirt store was chosen as the focus for this pilot study.

When people are shopping in the real world, they are experiencing it with all five senses - smell, sound, touch and hearing as well as vision. When shopping online their experience is primarily visual. By using probes in this study, researchers hoped to gain an insight into the stimuli that affect users when they are shopping and how their emotional reactions might influence the consumer decision process.

Most traditional Cultural Probe studies encourage users to take photographs or make drawings to express their feelings. In the given context, it was considered how users might feel walking around and taking photographs of things when they are shopping and whether this might disrupt their routine or make them feel awkward. They might struggle to take photos in a shop or even forget to take the camera with them. It was therefore necessary to devise a less obtrusive and more convenient approach. The participants were asked to use their own camera phones to take the pictures as they shopped. By doing so, they could subtly pretend they were texting or making a call and appear less conspicuous, as this technology is generally ubiquitous. A preliminary literature review indicated that several researchers have previously applied similar techniques with some success, naming them D*igital Cultural Probes* [6] (using mobiles to gather photos and audio clips from children to create pervasive digital technology) and M*obile Probes* [7] (based on two studies - using mobiles to gather data to develop a sales point for clothing retailers and a mobile work research study).

The apparent lack of literature in this subject domain indicated that it would be beneficial to explore the application of Digital Cultural Probes further and to evaluate and disseminate their appropriateness for future research projects.

For the purposes of this pilot exercise, three shoppers aged between 22 - 34, were asked to take photos and text them to the researcher. M1 was a male aged 34 years F1 and F2 were both female, aged 22 and 30 years respectively. M1 works as a self-employed carpenter, F1 is a nursery worker and F2 a traveller liaison officer. All three live and work in the Bristol area.

Most of the photos were taken over the course of three days at shops in Bristol. In the spirit of Gaver's original premise, the questions asked of the participants were not specifically focussed on t-shirt shopping and were rather ambiguous - leaving them open for interpretation and

unpredictable results. The participants were asked to photograph things related to their shopping experience in general, as well as clothes shopping. As this was only a pilot study to explore the appropriateness of the requirements elicitation techniques, the sample was quite small and only a short time frame was allocated in order to gather data.

Each participant was given written directions and asked to photograph things using their own mobile phone:

- What makes you want to go shopping? - text GO and a picture
- What might stop you from going shopping? - text STOP and a picture

- What would make you buy something? - text BUY and a picture
- What would stop you from buying something. - text DON'T BUY and a picture

- What makes you smile when you're shopping? - text SMILE and a picture
- What makes you frown when you're shopping? - text FROWN and a picture

All participants also received a short instructional text as a reminder which they could refer to when taking the pictures, rather than carrying a scrap of paper around. They were instructed that they could add a few words to the texts that they sent if they felt that an explanation was needed.

It was hypothesised that by texting their results, users would put the first thing that came into their head, without taking too much time to think about it. It was hoped that this would enable a truer snapshot of their emotions at that time, also denying them the chance to edit or re-evaluate their decisions.

It was also anticipated that allowing the users to participate in and in some respects, guide the requirements gathering process in this way would yield unexpected results, helping designers gain a deeper insight into facets of people's lives, which might not have been otherwise considered.

3.2 Sample Results

The participants sent a total of 37 texts, F1 was the most prolific, sending 17 texts. The request which prompted the most response was 'BUY' with a total of 13 responses from all participants. This could be due to the fact that 'buying' is the primary activity of their shopping excursion, but the experimental nature of this small sample precludes deeper analysis. On the whole, people preferred to report positive things rather than negative. All participants reported enjoying the experience, particularly the challenge of trying to take the photos without being detected. Here are some sample images and texts from some of the categories:

Fig. 2. BUY: Comment from user: "Clear layout"

Fig 3. BUY: Comment from user: "All these pretty colours together make me want to buy one in every colour. can't afford it though :("

Fig. 4. (previous page) DON'T BUY: Comment from user: "They look wrong. LOL! I would not shop here"

Fig. 5. STOP: Comment from user: "I'd rather go fishing"

4 Discussion

4.1 Benefits

The immediacy of these Digital Cultural Probes offers an effective method for capturing a user's innermost thoughts, on the move. They become active, creative contributors to a project, rather than being passively led through interviews or focus groups.

Looking at the examples given here, different kinds of information are offered by the users which can be interpreted or acknowledged in different ways.

In Fig. 2 a clear layout such as these books on a shelf may be directly translated into a system design. Would a user have contemplated offering this information in an interview or focus group and would they have been able to describe it as clearly without using a photograph? Probes offer a unique way to prompt users to consider their actions and thought processes in context, eliciting information which might otherwise go unnoticed.

In Fig. 3. this user shows another layout that draws the eye but muses on a hypothetical situation - they would like to buy all of the watches but could not afford to. Rather than specifically informing the design it is telling a story about the user - about their aspirations and how they feel at that moment when looking at those objects - a fleeting desire rather than a motivating influence which helps to build up a deeper picture of the user rather than a quantifiable result.

Fig. 4. Encapsulates the individuality and subjective qualities of the shopping experience. Users are responding to things in their environment. The probe captures personal thoughts, mood and feelings as they happen.

Fig. 5. This participant would rather be fishing than shopping. Understanding that some users are reluctant, and shopping under duress encourages an exploration of their motivation. Does this mean they would prefer to shop online than in the real world or would they dislike it equally? Perhaps developers should focus on saving time within a process so that users can find things quickly, buy them and leave, rather than spending a long time browsing on a site. The development of mobile devices means they could be shopping *whilst* fishing. Fig. 5. might prompt researchers to question further, conducting an in-depth interview with this user, prompted by their probe answers, to develop a deeper understanding of the problem and possible solutions.

By using text as a method of transmission, researchers are able to gather the information as it happens, with time and date included. The expediency of this method means that users are considering everything around them, not just the things that they recall afterwards. Mundane things such as flooring or parking spaces were reported as motivators or inhibitors to shopping; enabling researchers to construct a far richer picture than they would get from interviewing alone. The time stamp provided by the text message confirms that they were provided during the shopping process, so they were not afforded too much time to think or edit their responses. Once the text is sent it cannot be stopped, which also prevents them from returning to edit or resubmit any information captured in context.

By using their own mobile phone, participants feel more comfortable. It is a natural object to carry and more discreet than a notebook or camera for gathering data in situ. The playfulness of Gaver's original Cultural Probes is preserved in this study and the users enjoyed the exercise, stalking around the shops, covertly taking snaps and texting.

4.2 Challenges

4.2.1 Subjective Interpretation

Some researchers have expressed concern regarding the efficacy of the Cultural Probe approach in relation to offering insights. The outcome is biased to a certain extent on the researcher's interpretation of what is returned by the participant. Whilst the resulting insights *can* act as a catalyst for new design considerations, several studies have also discovered value in conducting follow-up interviews with participants to discuss their rationale or motives for submitting particular items, thus

offering greater opportunity for conducting further in-depth, participant-focussed interviews.

4.2.2 Preserving playfulness

In terms of developing this method for further studies, researchers must take steps to ensure that they preserve the sense of fun, acknowledging it's role both as an intrinsic motivator and a method for developing systems which are pleasurable to use. This may prove more difficult if carried out over a longer time frame or in more complex environments. The challenge of the Digital Probe designer is to make the process playful and engaging for the users whilst also being aware of their preferences.

4.2.3 Appropriateness and contexts of use

If a study is focussed on older adults or people who are not comfortable with mobile technology then it would be better to stick to lo-fidelity data gathering methods such as postcards or journals - whatever the user feels most comfortable with and enjoys doing the most and is most appropriate for a given situation. In this pilot study where the users were aged 22-34 mobile phone use is fairly ubiquitous and so it could be considered more appropriate as a method for in-context data collection.

Iversen and Nielsen's D*igital Cultural Probe* study worked with children in a research project to develop pervasive educational technology. They suggest this method was successful with this age group as it enabled access to the children's everyday lives, which could not easily be reached through other methods. The children's mobile phones were considered to be key artefacts and their familiarity and accessibility encouraged spontaneity amongst participants. The researchers applied their findings concerning the children's informal activities to cultivate in-depth interviews, resulting in a detailed collection of cultural data.

Hulkko et al.'s *mobile probe* study used probe software that had been developed specifically for the mobile platform combined with special software for receiving the data. Looking beyond photographs and texts, Gaver's original study used *psychogeographical* maps to represent areas (inspired by the provocative art of the situationalists). Smart phones could use maps for logging geolocation information in this way - perhaps prompting users to describe their feelings about certain places or simply to provide data in context. If a user is sending a picture from a shop, the data could also include the precise location of the photograph as well as the time and date. Of course not everyone has a smart phone and this would have to be considered when developing this method - they might not be as widely used by certain user sets and unfamiliarity with the technology could disrupt or inhibit the data collection process. Iversen and Nielsen applied digital sound recording snippets to their probes, which could also prove useful, depending on the context.

5 Conclusions

This pilot study is intended as an exploration of the application of Digital Cultural Probes. It is acknowledged that the sample size is too small to draw rigid conclusions about the data - the exercise was devised as an investigation into the value of using this method for further studies, based on preliminary findings and the small amount of available literature.

Initial findings suggest that this is a valuable method of enriching the data set, providing a unique opportunity for users to participate in the design process by providing researchers with access to their thoughts and feelings in context. By giving users a locus of control and creativity through participation, material generated from the probes can also help make participants feel more involved in the design process, as direct contributors of specific material which then becomes part of the project artefact collection.

The methods adopted here support Gaver's initial intention for Cultural Probes - encompassing playfulness and encouraging subjective engagement to promote empathy and deeper understanding of a situation. Researchers should accept that the results are not intended to be fully justified or quantified, and that their pervasive sense of uncertainty should be embraced as positive a value in the system design, as a tool for enhancing data or prompting further exploration.

We now aim to conduct further studies using Digital Cultural Probes using a much larger sample size, with a view to exploring in more depth, aspects such as the relationship between gender, culture and socio-economic attributes, on perspectives and needs.

In future, a larger study of this nature might benefit from applying prompts to the users to respond at certain times. Surprising them with tasks supports the original playful premise of the probes, motivating them to participate. As well as documenting current thoughts and feelings Digital Cultural Probes could also assist in indicating trends in new technologies - as social innovations and practices emerge within a new domain, such as mobile applications. There is certainly a lot of scope for them to be developed both as a method for enriching data and for facilitating deeper exploration during the application of traditional requirements gathering methods.

6 References

[1] Gaver, W.W., Dunne, A., & Pacenti, E. Cultural Probes. *Interactions* VI (1): 21–29. 1999

[2] Mayo, E. The Human Problems of an Industrial Civilization, New York: MacMillan, 1933.

[3] Gaver, W.W., Boucher, A., Pennington, S. & Walker, B. Cultural Probes and the Value of Uncertainty *Interactions* Volume XI (5): 53-36, 2004

[4] Boehner, K., Vertesi, J., Sengers, P., and Dourish. P., 2007. How HCI interprets the probes. In Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '07). ACM, New York, NY, USA

[5] Graham, C,. Rouncefield, M., Gibbs, M., Vetere, F. & Cheverst, K. How Probes Work *Proc. OzCHI Adelaide, Australia. 29-37* 2007

[6] Iversen, O. S. & Nielsen, C. Using Digital Cultural Probes in Design with Children *Proc. 2003 conference on Interaction design and children, July 01-03, Preston, Lancs, UK* 2003

[7] Hulkko, S., Mattelmäki, T., Virtanen, K. & Keinonen, T., Mobile Probes. In *Proc. NordiCHI, ACM Press: Tampere, Finland.* 2004

Requirements and Software Engineering for Tree-based Visualisation and Modelling - a User Driven Approach

Peter Hale, Tony Solomonides and Ian Beeson

University of the West of England, UWE

Abstract

This paper is about potential to provide an interactive visual ontology/taxonomy based modelling system. The research is part of efforts to structure, manage, and enable understanding of complex engineering, business and/or scientific information to enable those involved to collaborate using a systems approach. The aim and objectives are to provide a taxonomy management system to close the link between requirements gathering and end-user modellers. The research is into modelling of product data structures. This research could be adapted to business process modelling, and biology taxonomy visualisation/representation. The modelling system was developed to assist decision support for problems such as wing and engine design. The methodology involves modelling using tree structured ontology based modelling. It is argued that visualising this structure enables improved Maintenance, Extensibility, Ease of Use, and Sharing of Information, and so enables better and more accessible modelling. This is achieved by uniting the software taxonomy structure with the structure of the domain to be modelled and visualised, and using Semantic Web technologies to link this with ontologies and to end-users for visualisation. This research assists with management of development, use, and re-use of software in order to make this an integrated process. The research brings together related fields of Semantic Web, End-User Programming, and Modelling, to assist domain expert end users.

1 Introduction

There are opportunities to improve visualisation and interactivity capabilities of existing ontology representation and to combine this with modelling/end-user programming. The methodology, tools, and techniques to aid this are evaluated and discussed in section 2.2, and 2.3. This approach is combined with Semantic Web techniques to enable automated structuring and management of information, and make it more accessible via the web. This is the basis of the PhD work examined in this paper, and developed for the approach of User Driven Modelling/Programming (UDM/P).

Management, structuring, and visualisation information enables representation of complex hierarchical problems such as product and process modelling. This makes possible gathering and enabling representations of such problems, and a unified generic approach to this kind of modelling and linking of models via the Semantic Web. The aim is that a taxonomy management system will enable use of information, and a methodology for its representation and contextualisation in varied interactive ways, according to what is most useful for particular people and types of information. Applications are software and systems engineering, process and design/manufacturing modelling, and phylogenetic (biology trees). What is common to all these problems is their tree-based nature, suitable to taxonomies/ontologies.

Section 2 examines the problem to be solved, the role of ontologies, modelling, visualisation and interaction, and translation to aid end-user programming. The Position of software tools investigated within a table is analysed, to develop a way of combining tools for a User Driven Modelling/Programming (UDM/P) approach.

Section 3 examines objectives for enabling better more adaptable modelling, maintenance, extensibility, ease of use, and sharing of information for diagrammatic modelling. A methodology for UDM/P is developed

Section 4 reflects on implementing the methodology outlined in section 3 and shows that the main necessity is a translation to enable better modelling via improved human to computer translation. The importance of unifying the various tools and techniques through an umbrella tool suitable to end-users is discussed.

Section 5 concludes that the research showed a way of enabling domain expert modelling/programming by unifying tools and techniques to match end-users' needs. Section 5.1 examines future work.

2 Review of Tree-Based Modelling

2.1 Problem Statement

Software development is difficult for users because of time constraints, responsibilities and roles of employees that do not include the task of software development, and the need for experience of and access to programming tools. For modelling with relationship trees it is possible to construct visualisation software for non-programmers, and so improve ease of use and limit code writing by automating the requirements to model translation.

A methodology is required for creation of systems to enable collaborative end-user modelling/visualisation. This methodology could be applied to engineering process and product modelling and to allow scientists to model, visualise and debate taxonomies/phylogenies. Thus it could be proved that the methodology is generic to tree-based problems. This paper concentrates on engineering modelling.

Many computer literate people are experts in domains that require tree-based visualisation or modelling, such as engineering product structures, business process models, and biological phylogeny trees. The aim of this research is to convert requirements into a model and so enable these computer literate users to model and visualise problems by minimising code writing. This is a User Driven Modelling/Programming (UDM/P) approach. Models can include calculation via linked equations as in a spreadsheet but visualising the whole structure. If no calculations are needed just the visualisation is provided.

Research in ontology, modelling, visualisation and interaction is examined for integration into UDM/P.

2.2 Ontologies

The infrastructure of this research is an ontology that can be visualised and edited. This is step 1 of a translation process to generate a modelling system.

Gruber [1] defines and explains ontologies and examines how agreement could be achieved for ontology terms. Gruber defines an ontology as, “An ontology is an explicit specification of a conceptualization. The term is borrowed from philosophy, where an Ontology is a systematic account of existence. For AI systems, what ‘exists’ is that which can be represented” Gruber goes on to explain design criteria for ontologies. Gruber uses an engineering case study to examine usefulness of an ontology to engineers, and others who make use of equations and values with standard units. Uschold [2] states that “there is nothing inherently good about being further along the semantic continuum” ... (towards more formal ontologies) ... “In some cases there will be advantages; in other cases there will not. What is good is what works.”

Horrocks [3] explains, “An ontology typically consists of a hierarchical description of important concepts in a domain, along with descriptions of the properties of each concept.” He discusses ontology languages and their role in assisting with interoperability. Huber [4] examines issues in ensuring people transfer their knowledge and suggests organisation’s culture is important in peoples’ resistance to this. If people can see and interact with their and each others’ models this helps mitigate that problem.

Berners-Lee and Fischetti [5] sum up the advantage of the Semantic Web over other languages, and use of RDF (Resource Description Framework) Semantic Web language, “The advantage of putting the rules in RDF is that in doing so, all the reasoning is exposed, whereas a program is a black box: you don’t see what happens inside it.” The Semantic Web uses relationships to relate information and people. This relationship structure is explained as a ‘web’, and Berners-Lee and Fischetti, explain that the term ‘web’ denotes collections of nodes and links where any node can be linked to any other node. Berners-Lee and Fischetti argue for collaborative interactivity, - ‘Intercreativity’. They explain, “the world can be seen as only connections, nothing else.”

McGuinness [6] provides a useful guide on how ontologies can assist in linking distributed data. McGuinness considers the role of markup languages in defining content to be machine readable. McGuinness encourages creation of web-based visual representations of information to allow people to examine and agree on information structures. This linking and connectivity is also explained by Uschold and Gruninger [7]. McGuinness cites a diagram by Berners-Lee [8], which is further developed by Berners-Lee [9]. The concept illustrated, linked with that of ontologies contains representations of the place of each language in a layered stack alongside the purpose of the language. Each layer has an interoperable open standard interface.

McGuinness [6] outlines 7 ways ontologies are used :-

1 controlled vocabulary.
2 site organization and navigation support.
3 expectation setting.
4 “umbrella” structures from which to extend content.
5 browsing support.
6 search support.
7 sense disambiguation support.

Berners-Lee [10] explains "Despite excitement about the Semantic Web, most of the world's data are locked in large data stores and are not published as an open Web of inter-referring resources. As a result, the reuse of information has been limited."

Figure 1 outlines positioning of software to decide where each tool fits in the translation methodology to be devised. The modelling tool Vanguard System was chosen for the DATUM modelling project [12] because it handles Units and uncertainty. Advantages for the PhD

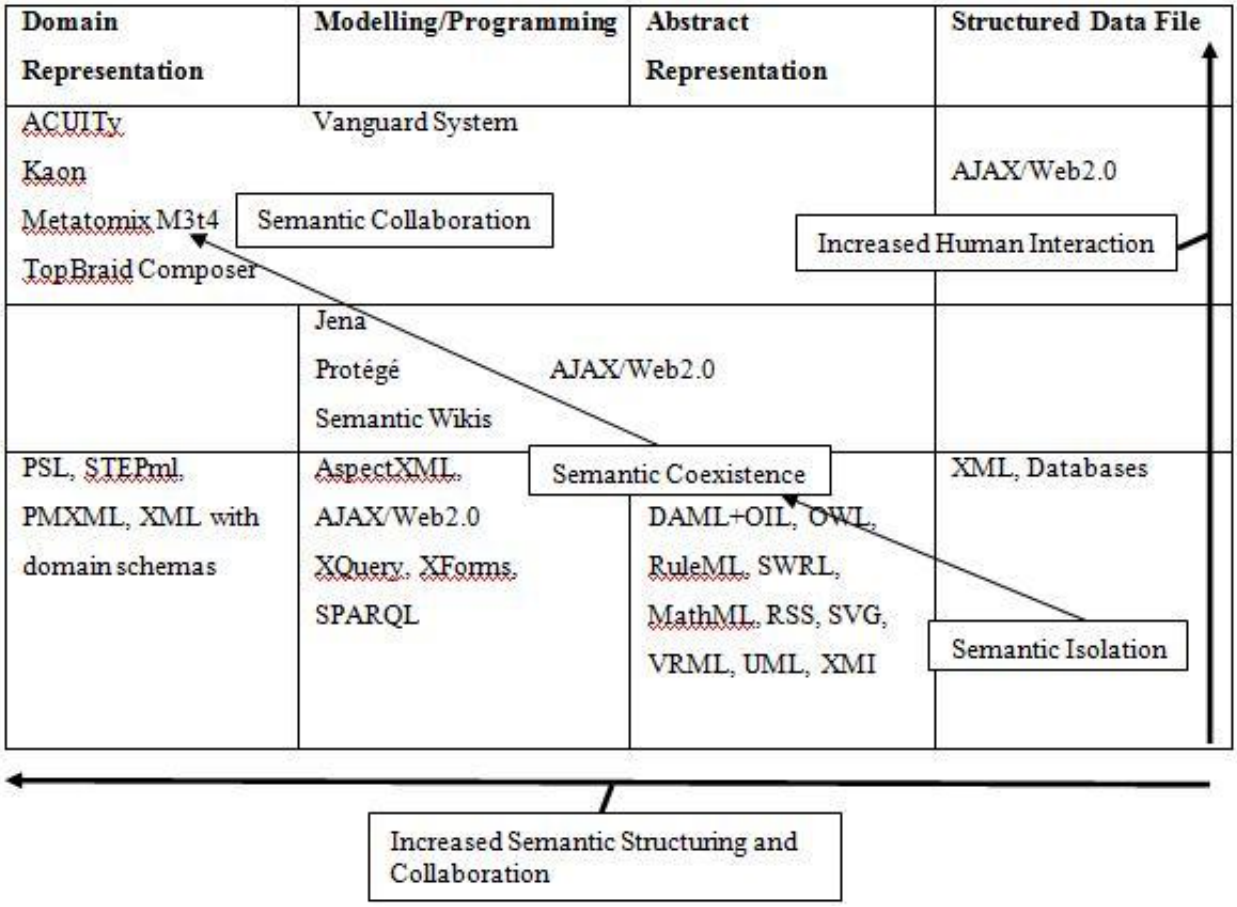


Figure 1. Language and Tool Mapping

Naeve [11] gives an example of the need for "semantic mapping" between different words with the same meaning such as 'author' in one ontology and 'creator' in another ontology in order to establish interoperability and machine readability. McGuinness [6] also investigates ontology tools/systems, and advocates their use for supporting collaboration for distributed teams. Naeve [11] describes Semantic Isolation where databases are available but hidden behind web portals, though the portals advertise their address. Semantic Coexistence is achieved by databases being structured in such a way that it is possible to search them without having to know their location. Naeve gives the example of RDF Schema RDF(S), which standardises the structuring of the information across RDF(S) databases. RDF(S) provides standardised elements for the description of ontologies, so assisting to enable Semantic mapping. Semantic mapping enables Semantic Coexistence by enabling agreement on terms. Naeve argues for semantics that are understandable to humans as well as machines; without that it is impossible for non programmer domain experts to undertake collaborative modelling.

were the facility to link to an ontology, collaborative and tree-based modelling capabilities, ease of use and of linking to spreadsheets and databases, facilities for web-based models, and for entering of formulae, and a high level programming language. The Protégé tree was translated into a Vanguard System tree. This fit in with the stepped translation to be developed. The open standard nature of Protégé made it possible to use it without being locked in. Tools such as TopBraid Composer provide additional higher level functionality such as an improved user interface and more facilities for user interaction and modelling by end-users. Leavers' MSc project [13] used Jena, and there was regular contact with the developers of ACUITy [14] to examine a Jena based approach. Jena based Metatomix M3t4 was also used. So results with Jena were similar to those for Protégé and Vanguard System. Analysing the position of tools within Figure 1 to ensure the best combination for a project is the most important way of choosing tools. The ontology tools all fit in the top quarter of the table and so provide similar functionality. Such ontology tools, and all the other tools in Figure 1 are improving, so reproducing this research by modelling at high level with involvement of end-users is thus practical.

2.3 Modelling

Ontologies are a base for modelling tools to provide a structured system for building and editing of models. An ontology can store related information and calculations; any required calculations would then be made and translated to provide a model that can be interpreted by users. This research solves problems of translation from human to computer and vice versa. This is achieved by giving users involvement in the translation process by providing for them to interactively model the problem.

Cheung et al. [15] wrote a useful guide to visualising and editing ontologies, this and interoperability via open standards languages makes modelling practical. Linking ontologies with modelling make them useful in engineering, and science, where calculations are required.

2.4 Visualisation and Interaction

Huhns [16] and Paternò [17] explain that alternatives to current approaches of software development are required. Huhns argues that current programming techniques are inadequate, and outlines a technique called 'Interaction-Oriented Software Development', concluding that there should be direct association between users and software, so users can create programs. Translation between ontologies, models, and visualisation enables translation between levels of abstraction, and therefore from human to computer and back. This approach concentrates on visualising the entire program code to end-users as a model. This is how to allow people to program modelling solutions at the level of abstraction they are most comfortable with. Paternò [17] outlines research that identifies abstraction levels for software systems.

Crapo et al. [18] argue that spreadsheet users are considered as potential modellers, "Every one of the perhaps 30 million users of spreadsheet software can be considered a potential modeller". Crapo et al. also explain that visualisation helps modellers to maintain a hierarchy of sub models at different stages of development and to navigate effectively between them. This is the reason for breaking down the models into a tree/graph/web structure. Jackiw and Finzer [19] and Guibert et al. [20] demonstrate how a view of the problem that is visual and near to peoples' way of thinking helps modellers. Context is essential. Guibert et al. explain with an example of a numerical representation of a triangle how numbers fail to reveal the concept behind them. This representation is 'fregean' as it does not show the concept of a triangle. Beside this is a diagram of the triangle that shows the concept, this is an 'analogical' representation as it includes the context of the information. Visualisation and interaction research enables end-user programming, such as for engineers to model/program at a high level of abstraction.

2.5 End-User Programming

End-user programming development over past decades was also reviewed. Two main conclusions resulted :-

- Research that created software for end-user programmers such as children, but had limited acceptance and use in the market can be reused with new technology to assist development.
- Pragmatic research that involved creation of tools for the mass market, but which avoided more long term issues can now be extended.

Section 3 develops the knowledge gained in section 2 into a methodology for tree-based programming.

3 Development of Methodology

3.1 Objectives

An objective is to develop a process to enable decision support, minimising dependence on specialist software and detailed programming. The User Driven Modelling/Programming (UDM/P) approach and its application to systems modelling research is developed.

This research examines creation of models and modelling systems, and how this can be eased for non-programmers. It identifies ways that creation of models and modelling systems is similar to other types of programming, so the research can be applied generally. The purpose is to enable end-users to create and adjust models and so maximise maintenance, extensibility, ease of use, and sharing of information; in order to develop a systematic methodology for creation of accessible and adaptable models, applicable to a range of situations. This enables end-users to model their domain problems.

3.2 Requirements

The development process investigated is that of ontology based translation between requirements, models and visualisation Figure 2 illustrates how this is most applicable to tree-based problems and models :-

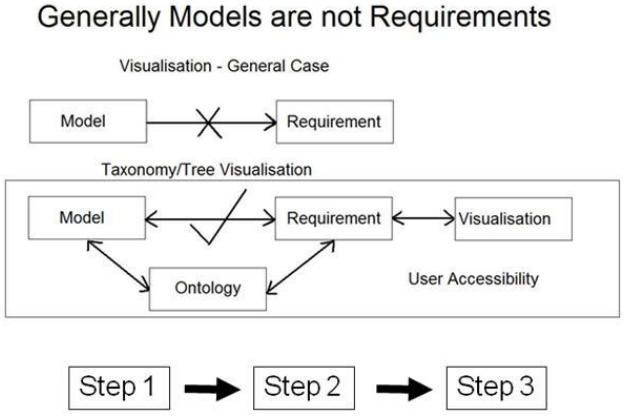


Figure 2. Visualisation and Interaction Mechanism

3.3 Methodology - Enabling Better and more Accessible Modelling

This methodology is used to enable better management of software through improved maintenance, extensibility, ease of use, and sharing of information. This makes software management an integral, consistent, continuous part of development, use and re-use.

Maintenance

Maintenance of models and modelling systems is improved by :-

- Stepped translation process, Step 1 Ontology and Taxonomy creation, Step 2 Translation and Modelling, Step 3 Translation and Visualisation.
- Use of open standards to represent information in a format available to the maximum range of maintainers without being dependent on the computer system or software used.
- Ensuring the structure of the modelling and programming system and all its related information is visualised clearly. This is ideal for point '5. *Maintainability*' Sommerville [21], this is also ideal for tree-based modelling systems.
- Minimising the amount of code necessary to create a model, and structuring the model so that all connections can be seen.

Extensibility

Extensibility is also improved by the above means; this enables understanding of models and so allows for easier re-use. A structured representation can be edited with fewer worries about unintended consequences. This is achieved by translation and visualisation to enable model builders and users to modify the ontology and model. This is the 3-step translation process developed for User Driven Modelling/Programming (UDM/P). Users make changes to whichever step is appropriate depending on the task they are performing and their interests and preferences. McGuinness [6] observes the importance of extensibility, "Extensibility. It will be impossible to anticipate all of the needs an application will have. Thus, it is important to use an environment that can adapt along with the needs of the users and the projects."

Ease of Use

- Maximising accessibility is important to ease of use and vice versa, use of open standards assists this, as does enabling models to run on software and systems familiar to users.
- Clear structuring and visualisation of information/models also assists in making a modelling system more usable.

Sharing of Information

Achievement of all the above enables collaboration. Ontologies are used as a way of representing explicit and implicit knowledge. This makes possible creation of manageable, maintainable, and flexible models. To enable sharing of information, diagrammatic ontology based representations of models are provided.

3.4 Translation

Translation capabilities are provided to enable better communication between computer systems, and between humans and computer systems. This allows visualisation of chains of equations, which are common in cost modelling, but this work is relevant to modelling in general. To model complex problems a structured approach is needed for representing explicit and implicit knowledge. A translation is provided in 3 steps :-

- Step 1 - Ontology
- Step 2 - Modelling Tool
- Step 3 - Interactive Visualisation

Step 3 visualises results and allows interaction with information to establish the meaning of results. The translation is based on Semantic Web standards to enable widespread applicability and help ensure this is a generic solution. The visualisation and interactions can be tree/graph-based, spreadsheet, and CAD style as necessary. A further alternative is translation to programming or Meta-programming languages so information can be re-used by developers who are creating systems with these languages.

In general it is likely that there will be merging between different modelling approaches and technologies. This needs organisation and management through an integrated system. UDM/P is thus an umbrella activity.

The standardisation possible in this approach allows software developers to create modelling systems for generic purposes that can be customised and developed by domain experts to model their domain. This methodology is facilitated by :-

- Modelling Tools - Building an end-user interface and extending the translation capabilities of UML and/or other modelling tools (Johnson, [22] - to be discussed in 4.3).
- Spreadsheets - Improving the structuring and collaboration capabilities of spreadsheets, and enabling customisation of spreadsheet templates for particular domains and users.
- Ontology Tools - Extending modelling capabilities and equation calculations in ontology tools and providing an end-user interface.

- Semantic Web/Web 2.0 - Extending the capabilities of Semantic Web and Web 2.0 style development tools to allow collaborative modelling.

These possible solutions are not mutually exclusive and their combination is the best way of providing usable collaborative modelling tools for computer literate end-users and domain experts. The link between these alternative ways of advancing current research is translation and User Driven Modelling/Programming (UDM/P).

Section 4 reflects on the prototyping, implications, advantages, and problems for this work.

3.5 Information management and Interaction

Figure 3 illustrates the development process. It shows how production of better, more accessible, more adaptable and applicable models was enabled by meeting objectives of enabling better Maintenance, Extensibility, Ease of Use, and Sharing of Information. These objectives were met by better structuring and visualisation; this required work on structuring using Semantic Web and Ontologies, and enabling better visualisation through end-user programming techniques. This made the models more accessible, and so easier to edit, reuse, adapt and maintain, so providing a more manageable development process.

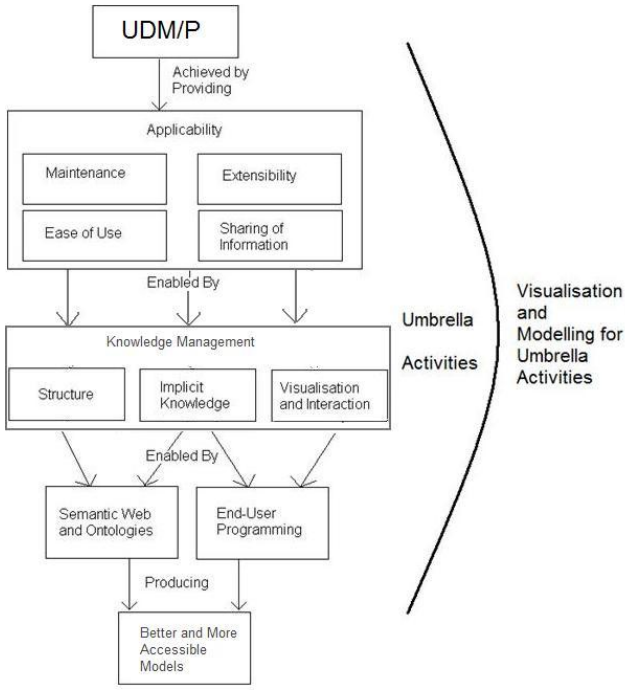


Figure 3. How Objectives and Methodology aid better modelling

Examination of this problem has indicated the need for management and co-ordination of a collaborative ontology, modelling, and visualisation process. This umbrella structure is required to manage the translation steps; in order to output accessible and better models.

3.6 Implementation of Methodology

Translation

To prototype and implement this methodology, an ontology representation was translated into a computer model. This ontology defined relationships between engineering items, the ontology was linked to Semantic Web technologies. The relationships were conveyed to a software model for evaluation. The taxonomy and CAD type view was then visualised and output to the web. The 3-step process methodology and implementation are illustrated in [23] and [24], and figure 4.

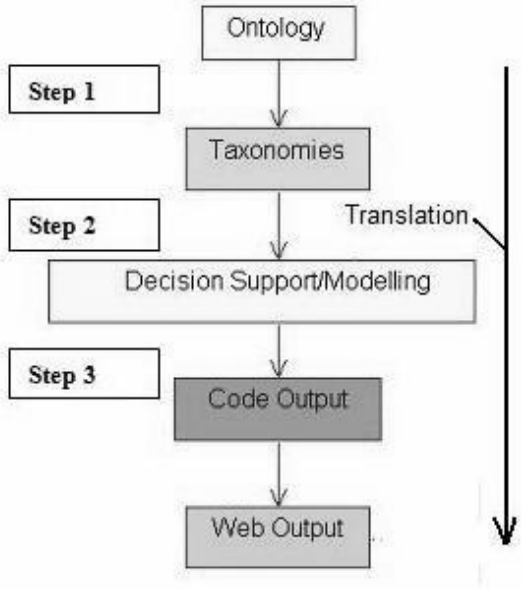


Figure 4. Translation Process Chain

4 Reflection and Discussion

This research assists with an integration of modelling, software engineering and systems engineering with a unified approach. This approach enables systems that produce systems, and models that produce models, systems that create models etc. This provides an iterative recursive translation, collaboration, and visualisation for modelling, thus improving capabilities for modelling.

This research closes the gap between developers and users, and between formal and less formal development processes. This is possible by providing an interface to model the process and the requirements and software structure in a visualised, interactive accessible way. If users drive the development process via accessible visualisation/modelling with high level modelling tools, this process then becomes agile and collaborative.

An overall modelling/visualisation structure would allow the user to establish “common ground” with the computer, an expression used by Johnson [22]. As well as

translating between users and computer systems it is important to provide translations between different computer systems. Solving this would enable providing a modelling and simulation environment as a product of translation from an ontology. Miller and Baramidze [25] establish that for a "simulation study that includes model building, scenario creation, model execution, output analysis and saving/interpreting results. Ontologies can be useful during all of these phases." Kim et al. [26] describe their approach to modelling and simulation and how a Web-based solution can be applied to distributed process planning. So a web-based ontology editor that enables modelling and visualisation is needed.

Naeve [11] argue that "combining the human semantics of UML with the machine semantics of RDF enables more efficient and user-friendly forms of human-computer interaction." The main difficulties that need to be addressed to enable this are structural differences between Semantic Web and UML representations, and the need for improved human interaction for non programmer users. Naeve examines the strong separation between types (classes), and instances (objects) and considers this to be a weakness, which he rectifies for ULM (Unified Language Modeling) developed from UML. Johnson [22] indicates that UML tools need extending to better enable modelling of collaborative tasks. Johnson explains that successful interaction requires mapping between levels of abstraction and that translation between the levels of abstraction required by users and computers is difficult. He explains that this problem often means systems are created that make the user cope with mis-translation. Fischer [27] observes that it is the mismatches between end-users needs and software support that enable new insights. Fischer argues that software development can never be completely delegated to software professionals, because domain experts are the only people that fully understand the domain specific tasks that must be performed.

To enable computer to human common ground, an interactive, visualised ontology/modelling environment is researched. This is adapted to the way people work, with steps matched to people, skills, and roles :-

Table 1. Roles, Skills, and Translation

Step	Person Role	Skills	Tool Type
Step 1	System Creator	Programmer	Ontology
Step 2	Model Builder	End-User Programmer	Modelling Tool
Step 3	Model User	End-User	Interactive Visualisation

This stepped translation solved problems as indicated in the table below :-

Table 2. Stepped Translation and Modelling

Improvement	Achieved By
Maintenance	Structuring and Translation
Extensibility	Structuring and Visualisation
Ease of Use	Visualisation, Interaction, and Translation
Sharing of Information	Shared Ontology and Interoperability

4.1 Recommendations

- Enable people to create software visually.
- Create design abstractions familiar to domain experts e.g. diagrams for engineers.
- Ensure interoperability using open standards.
- Automate user to computer translation process.

5 Conclusions

This translation and management process was adapted to match with relevant tools, roles and skills to provide a framework for UDM/P modelling. Software tools were combined and used for end-user needs and the UDM/P approach. This enables interactive modelling and visualisation, and so widens programming participation by including computer literate non-programmers. The main ways to achieve this are through better models provided by means of improved Maintenance, Extensibility, Ease of Use, and Sharing of Information.

An issue is whether and how and by whom such an approach can be moved out of University and into industry in a practical way. This is especially difficult given the more short term pressures facing businesses/organisations. A further issue is that this approach does not suit rigid hierarchical organisations, despite being based on a hierarchical structure itself. The approach involves empowerment of users. This means it is important to enable collaboration across people, and up and down the model hierarchy. Thus this supports a democratic, decentralised structure and enables this.

5.1 Future Work

Given an analysis of Proctor et al. [28] there is a gap in research in creation and editing of web-based trees. Future work post PhD will involve improving capabilities of modelling information, such as Semantic Web technologies combined with development of increasingly interactive programmable web interfaces. This could help make possible Tim Berners-Lee's [8][9] original vision of Web 3.0 that involves structured information linked via a stack of technologies, each providing a layer of Semantics above the layer below, to provide a computer to human translator.

References

[1] T. R. Gruber, Toward Principles for the Design of Ontologies Used for Knowledge Sharing. N. Guarino and R. Poli, ed. *Formal Ontology in conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers, 1993.

[2] M. Uschold, Ontologies Ontologies Everywhere - but Who Knows What to Think? *9th Intl. Protégé Conference, Stanford, California*, July 23-26, 2006.

[3] I. Horrocks, DAML+OIL: a Reason-able Web Ontology Language. *Proceedings of the Eighth Conference on Extending Database Technology (EDBT 2002)*, March 24-28 2002.

[4] G. P. Huber, Transfer of knowledge in knowledge management systems: unexplored issues and suggested studies. *European Journal of Information Systems*. 10:80-88, 2001.

[5] T. Berners-Lee and M. Fischetti, *Weaving the Web*. Harper San Francisco; Paperback: ISBN:006251587X, 1999.

[6] D. L. McGuinness, Ontologies Come of Age. In: D. Fensel, J. Hendler, H. Lieberman, and W. Wahlster, ed. *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*. MIT Press, 2003.

[7] M. Uschold, M. Gruninger, Ontologies and Semantics for Seamless Connectivity. *Association for Computer Machinery - Special Interest Group on Management of Data, SIGMOD Record* December, 33(4), 2004.

[8] T. Berners-Lee, *Semantic Web on XML - Slide 10*, http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html, 2000.

[9] T. Berners-Lee. *Semantic Web "Layer Cake"*, http://www.w3.org/2004/Talks/0412-RDF-functions/slide4-0.html, 2004.

[10] T. Berners-Lee, W. Hall, J. Hendler, N. Shadbolt, D. J. Weitzner, Creating a Science of the Web. *Science 11*, 313(5788):769- 771, 2006.

[11] A. Naeve. The Human Semantic Web - Shifting from Knowledge Push to Knowledge Pull. *International Journal of Semantic Web and Information Systems (IJSWIS)* 1(3):1-30, 2005.

[12] J. Scanlan, A. Rao, C. Bru, P. Hale, R. Marsh, DATUM Project: Cost Estimating Environment for Support of Aerospace Design Decision Making. *Journal of Aircraft*, 43(4), 2006.

[13] N. Leaver, Using RDF as an Enabling Technology. MSc. Dissertation, University of the West of England, Bristol, 2008.

[14] A. Aragones, J. Bruno, A. Crapo, M. Garbiras,. An Ontology-Based Architecture for Adaptive Work-Centered User Interface Technology. *Jena User Conference*, 2006, Bristol, UK.

[15] W. M. Cheung, P. G. Maropoulos, J. X. Gao, H. Aziz, Ontological Approach for Organisational Knowledge Re-use in Product Developing Environments. *11th International Conference on Concurrent Enterprising - ICE*, 2005.

[16] M. Huhns, Interaction-Oriented Software Development. *International Journal of Software Engineering and Knowledge Engineering*, 11:259-279, 2001.

[17] F. Paternò, Model-based tools for pervasive usability. *Interacting with Computers*, 17(3):291-315, 2005.

[18] A. W. Crapo, L. B. Waisel, W. A. Wallace, T. R. Willemain, Visualization and Modelling for Intelligent Systems. C. T. Leondes, ed. *Intelligent Systems: Technology and Applications, Volume I Implementation Techniques*, 3:53-85, 2002.

[19] R. N. Jackiw, W. F. Finzer, The Geometer's Sketchpad:Programming by Geometry. A. Cypher, ed. *Watch What I Do: Programming by Demonstration. MIT Press*, Chapter 1, 1993.

[20] N. Guibert, P. Girard, L. Guittet, Example-based Programming: a pertinent visual approach for learning to program. *Proceedings of the working conference on Advanced visual interfaces*. 358-361, 2004.

[21] I. Sommerville, Software Engineering, *Eighth Edition. Addison-Wesley*, 242-243, 2007.

[22] P. Johnson, Interactions, collaborations and breakdowns. *ACM International Conference Proceeding Series; Proceedings of the 3rd annual conference on Task models and diagrams*, 86 Prague, Czech Republic, 2004.

[23] P. Hale, A. Solomonides, I. Beeson. User-Driven Modelling: Visualisation and Systematic Interaction for end-user programming, *Journal of Visual Languages and Computing*, To be published.

[24] P. Hale, User Driven Modelling: Visualisation and Systematic Interaction for End-User Programming. *PhD*, UWE, To be published 2011.

[25] J. A. Miller, G. Baramidze, Simulation and the Semantic Web. *Proceedings of the Winter Simulation Conference*, 2005.

[26] T. Kim, T Lee, P. Fishwick, A Two Stage Modeling and Simulation Process for Web-Based Modeling and Simulation. *ACM Transactions on Modeling and Computer Simulation*, 12(3):230-248. 2002.

[27] G. Fischer, Meta-Design: A Conceptual Framework for End-User Software Engineering. *End-User Software Engineering Dagstuhl Seminar*, 2007.

[28] J. B. Proctor, J. Thompson, I. Letunic, C. Creevy, F. Jossinet, G. J. Barton, Visualizing of multiple alignments, phylogenies and gene family evolution, *Supplement on visualizing biological data*, March 7(3), 2010.

A Prediction Model for the Stability Factor of Safety in Soil Slopes

A. Ahangar-Asr*, N. Mottaghifard, A. Faramarzi, A. A. Javadi

Computational Geomechanics Group, School of Engineering Mathematics and Physical Sciences, University of Exeter

**aa375@exeter.ac.uk*

Abstract

Analysis of stability of slopes has been the subject of many research works in the past decades. Prediction of stability of slopes is of great importance in many civil and geotechnical engineering structures including earth dams, some retaining walls, trenches, etc. There are several parameters that contribute to the stability of slopes. In this paper a new approach is presented based on evolutionary polynomial regression (EPR) for modelling of stability of slopes. EPR is an evolutionary data mining technique that generates a transparent and structured representation of the behaviour of a system directly from data. Like neural networks, this method can operate on large quantities of data in order to capture nonlinear and complex interactions between variables of the system. It also has the additional advantage that it allows the user to gain insight into the behaviour of the system. EPR model is developed and validated using results from sets of data from literature. The main parameters contributing to the behaviour of slopes namely, unit weight, cohesion, friction angle, slope angle, and pore water pressure are used in the development of the EPR model. The developed model is used to predict the factor of safety of slopes against failure for conditions not used in the model building process. The results show that the proposed approach is very effective and robust in modelling the behaviour of soil slopes and provides a unified approach to analysis of slope stability. The merits and advantages of the proposed approach are highlighted.

1 Introduction

Traditional limit equilibrium techniques are the most commonly used methods for analysis of stability of slopes. In this approach, the shape and location of the critical failure surface are assumed rather than determined. It is also assumed that the soil (or rock) moves as a rigid block with movements only occurring on the failure surface. The factor of safety (F_S) is defined as the ratio of reaction over action, expressed in terms of moments or forces, depending on the mode of failure and the geometry of the slip surface considered. In rotational mechanisms of failure for example, factor of safety is defined, in terms of moments about the centre of the failure arc, as the ratio of the moment of the resisting shear forces along the failure surface over the moment of weight of the failure mass. These computational methods vary in terms of degrees of accuracy, depending on the suitability of the simplifying assumptions for the situation under investigation. In recent years, the use of artificial neural network has been introduced as an alternative approach for analysis of stability of slopes. Sakellariou and Ferentinou [1] have applied the neural network procedure to acquire the relationship between the parameters involved in analysis of the stability of slopes. They have used the models introduced by Hoek and Bray [2] in order to produce test data to validate the quality of the learning process of artificial neural networks. In this paper a new approach is introduced for modelling the behaviour of soil slopes, which integrates numerical and symbolic regression to perform evolutionary polynomial regression (EPR). The key idea behind the EPR is to use evolutionary search for exponents of polynomial expressions by means of a genetic algorithm (GA) engine.

2 Methods of Slope Stability Analysis

Slope stability is usually analysed by limit equilibrium methods. These methods were developed before the advent of computers; computationally more complex methods appeared later. These methods require information about the strength parameters and the geometrical parameters of the soil or rock mass. In rock

mass, the potential mechanism of failure can be wedge or planar, depending on the orientation of joint sets, in soils and highly fractured rocks this mechanism can be circular.

The methods introduced by Taylor [3] and Bishop [4], assume a circular shape of slip surface. Other methods introduced by Janbu [5], Spencer [6], Sarma [7] and Hoek and Bray [2], assume any shape of slip surface. The accuracy of the above-mentioned methods depends on their assumptions and the accuracy with which shear strength can be measured.

3 Some Features of EPR

EPR is a technique for data-driven modelling. In this technique, the combination of the genetic algorithm to find feasible structures and the least square method to find the appropriate constants for those structures implies some advantages. In particular, the GA allows a global exploration of the error surface relevant to specifically defined objective functions. By using such objective (cost) functions some criteria can be selected to be satisfied through the searching process. These criteria can be set in order to (a) avoid the over fitting of models; (b) push the models towards simpler structures; and (c) avoid unnecessary terms representative of the noise in data. EPR shows robustness and in every situation can get a model truly representative of data. An interesting feature of EPR is in the possibility of getting more than one model for a complex phenomenon. A further feature of EPR is the high level of interactivity between the user and the methodology. The user physical insight can be used to make hypotheses on the elements of the target function and on its structure. Selecting an appropriate objective function, assuming pre-selected elements based on engineering judgment, and working with dimensional information enable refinement of final models. Detailed explanation of the method can be found in [8, 9].

4 Data base

The main goal of this research is to implement the above methodology in the problem of slope stability estimation. In order to introduce a model for slope stability analysis to predict the factor of safety (F_S) or the status of stability (S) in soil slopes, the factors that influence F_S and S have to be determined. The output EPR model is a single polynomial equation for the factor of safety (F_S) which demonstrates the status of stability of the slope (S).

Two data sets consisting 46 and 21 case studies of slopes analysed for circular critical failure mechanism [1] are used. All cases are dry (8 failed, 7 stable) [10]. These data cover a wide range of parameter values.

The parameters that have been selected are related to the geotechnical properties and the geometry of each slope. More specifically, the parameters used for circular failure (Figure 1) were unit weight, cohesion, angle of internal friction, slope angle, height, and pore water pressure.

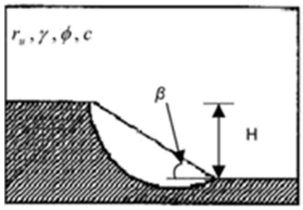


Fig. 1 Circular failure mechanism

5 Circular Failure Mechanism

From 67 total cases of data 10 cases are chosen to be used as unseen data cases to validate the EPR developed model and the remaining sets are used in the training stage.

Among 8 resultant equations developed by EPR the one with the highest COD value (75.99%) is selected (Equation 1).

$$F_S = -\frac{1.49 \cdot H}{\gamma^2} - 1.8 \cdot r_u^2 + 2.59 \cdot \tan(\phi) - 2.18 \cdot \tan(\phi) \cdot \tan(\beta) + 0.014 \cdot C - 5.19 \times 10^{-5} \cdot C^2 + 0.817 \quad (1)$$

Figure 2 shows the comparison of the resulted factors of safety from EPR model with the ones from ANN analysis [1] and the experimental data for training data. A great consistency between EPR, ANN, and experimental data can be easily seen.

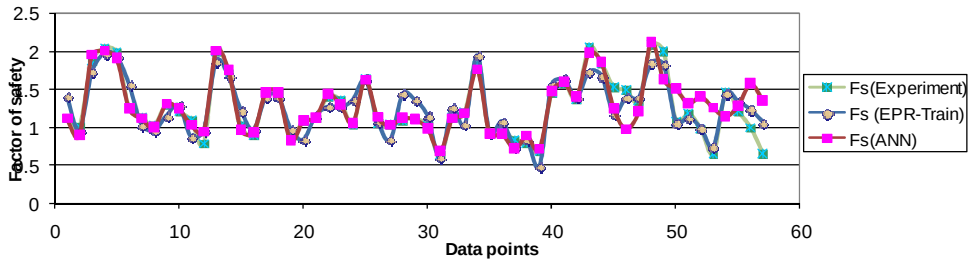


Fig. 2. Comparison of EPR training results with ones from ANN and experimental data for factor of safety in circular failure mechanism

Once training is completed, the performance of the trained EPR model is validated using the unseen validation data. The purpose of the validation is to examine the capabilities of the trained model to generalise the training to conditions that have not been seen during the training phase. Equation 1 is used to predict the factor of safety for unseen data cases and results are shown in figure 3. A very good agreement can be seen between the model results and the laboratory experimental data demonstrating the great capability of the EPR based model in generalising the relationship for unseen data cases.

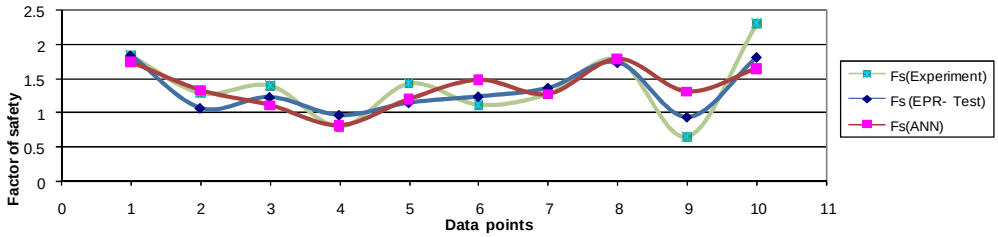


Fig. 3. Comparison of EPR testing results with ones from ANN and experimental data for factor of safety in circular failure mechanism

6 Summary and Conclusion

In this paper, a new approach was presented, based on evolutionary polynomial technique, for the analysis of stability of slopes. An EPR model was trained and validated using a database of case histories involving stability status of slopes. The results were compared with data obtained from field as well as artificial neural network results.

Comparison of the results shows that the results predicted by the proposed EPR model provide a noticeable advantage over the ANN model; it introduces a polynomial constitutive equation which gives a very clear picture of the relationship between the contributing parameters. It is shown that the model is capable of learning, with a considerably high accuracy, the underlying relationship between the factor of safety of the slopes and the contributing factors in the form of a polynomial function and generalizing the training to predict factors of safety for new cases. The results show that the EPR model provides a very efficient way for the stability analysis of soil slopes.

References

[1] M. G. Sakellariou, and Ferentinou, M.D., "A study of slope stability prediction using neural networks," *Geotechnical and Geological Engineering*, vol. 23, pp. 419-445, 2005.

[2] E. Hoek, Bray J.W., *Rock Slope Engineering*, 3 ed. London: Institution of Mining and Metallurgy, 1981.

[3] D. W. Taylor, "Stability of earth slopes," *J. Boston Soc. Civil Eng.*, vol. 24, pp. 197, 1937.

[4] A. W. Bishop, "The use of slip circle in the stability analysis of earth slopes," *Geotechnique*, vol. 5, pp. 7-17, 1955.

[5] N. Janbu, "TitlConference Proceedingse," presented at Fourth European Conference on stability of earth slopes, 1954.

[6] E. Spencer, "A method of analysis of the stability of embankments assuming parallel inter-slice forces," *Geotechnique*, vol. 17, pp. 11-26, 1967.

[7] S. K. Sarma, "Seismic stability of earth dams and embankments," *Geotechnique*, vol. 25, pp. 743-761, 1975.

[8] D. Savic, Giustolisi, O., Berardi, L., Shepherd, W., Djordjevic, S., and Saul, A., "TitlConference Proceedingse," presented at Proceedings of ICE, Water Management, 2006.

[9] M. Rezania, Javadi, A.A., and Giustolisi, O., "An Evolutionary-Based Data Mining Technique for Assessment of Civil Engineering Systems," *Journal of Engineering Computations*, vol. 25, pp. 500-517, 2008.

[10] N. K. Sah, Sheorey P.R., and Upadhyama L.W., "Maximum likelihood estimation of slope stability," *Int. J. Rock Mech. Min. Sci. Geomech. Abstr.*, vol. 31, pp. 47-53, 1994.

Integrating Design for Manufacturability with CAD for the Grinding of Tapered Rollers

Dr. Craig Seidelson

Chief Engineer Manufacturing Advancement (Timken China)
Department of Environment and Technology Univ. of the West of England

Abstract

Computer aided design and drafting (CADD) involved defining a product before it was ready to be manufactured. In the bearing industry, CADD definitions started with 2 or 3 dimensional representations of bearing components. Designers increasingly used computers to then model such features as bearing assembly clearances and deformations under load.

Computational modeling, however, did not extend to bearing manufacture. For example, those designing tapered rollers had no idea if the geometries selected predetermined roller radial deviation from round (OOR) and productivity.

For the through feed grinding of tapered rollers, the researcher discovered it was possible to use roller length, nominal diameter, and taper angle to predict maximum productivity. The 3 roller geometry factors, in combination, were inputs to a negative sloped linear regression. The researcher, likewise, discovered it was possible to use roller nominal diameter to predict OOR. Nominal diameter was an input to a positive sloped linear regression.

By integrating regression findings with CADD, the researcher produced a novel "Tapered Roller DFM" program. The software allowed bearing designers to simultaneously evaluate 4 roller designs for optimum manufacturability and energy efficiency. For example, per the roller length, taper, and nominal diameter selections, grind productivity and manufacturing costs were displayed. Per roller nominal diameter selections, OOR and application power losses due to rotational displacements were presented. To ensure roller designs met application requirements, associated bearing speed and load limits were likewise displayed.

1 Introduction

Today's Computer Aided Drafting (CAD) platforms offered designers such features as: (a) automated calculation, (b) fit, cut, and force displacement modeling, (c) rapid revision and scaling, (d) download capability to program logic controllers, and (e) collaboration across databases [1]. In the tapered roller bearing industry, the researcher observed CAD users leveraging features (a-e) to design the next generation of fuel efficient bearings. However, at the research sponsor, Timken, CAD users did not consider whether tapered roller geometry impacted application energy efficiency and/or roller manufacturability. To quantify the impact of ignoring Design for Manufacturability (DFM), a work piece's design, on average, was found to determine 70% of its manufacturing cost [2].

1.1 Scope

The researcher theorised in the through feed (i.e. traverse) centreless grinding of tapered rollers, 3 roller geometry factors predetermined productivity and application energy efficiency. The 3 roller geometry factors tested (Figure 1-1) were: length (R_L) (because of its affect on grinding feed rate), taper (β), and nominal diameter (D_{NR}). D_{NR} and β affected grinding set up angles ϕ_1 and ϕ_2.

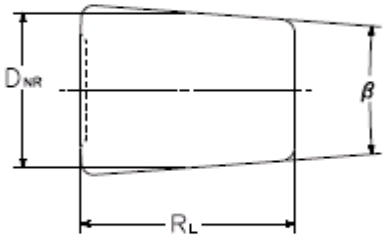


Figure 1-1 Tapered Roller Geometry Factors

Productivity was defined as maximum roller rev/sec during grinding. Energy inefficiency was defined as

bearing power loss due to roller out of roundness (i.e. "OOR").

A proposed connection between grinding angles, feed rate, and productivity was based on rotational instability work done by Barrenetxea et al. [3]. A proposed connection between grinding angles, feed rate, and OOR was based on Dall's kinematics work [4] and Snoeys and Brown's [5] dynamics work. Power loss increased with roller OOR because the force of imbalance displacing a rotating mass on a rigidly mounted bearing shaft increased [6] per Equation 1-1. As rotational imbalance increased so too did power loss [7] per Equation 1-2.

$$F(N) = m(kgs)\omega^2(\frac{rads}{\sec})OOR(m) \qquad (1\text{-}1)$$

$$P(W) = F(N)V_m(\frac{m}{\sec}) \qquad (1\text{-}2)$$

The researcher, also, theorised linear regression equations existed which related the 3 roller geometry factors with grind productivity and application energy efficiency. Using these equations the researcher sought to write a computer program providing the CAD user with Web-based design review.

2 Literature Review

Timken, the largest tapered roller bearing supplier to the US automobile industry [8], turned to academia to expand its CAD capabilities to include DFM. A linkage between academia and the automobile industry in computer systems development was consistent with the origins of CAD. For example, Sutherland's "Sketchpad" software, developed at MIT as part of his PhD thesis in the early 1960's, was the first to allow users direct interaction with computer. The method was a light pen tracing on the computer monitor [9]. Hanratty's "PRONTO" numerical control software, developed at General Motors Research Laboratories in 1957, was the first industrial CAD application [10]. The purpose was tool design. Just as General Motors and MIT's Mathematical Laboratory led the way in 2-D CAD development, subsequent 3-D CAD packages, in the late 1960's, originated at Cambridge Universities Computing Laboratory and automotive manufacturers Citroën and Renault [10].

The fact the automobile industry played an early role in CAD development was due its large scale engineering needs and ability to pay the high computing costs. For example, one of the first commercially available CAD systems (i.e. Digigraphics) cost upwards of $500,000 [10].

In 1971, Hanratty wrote an original drafting package called Adam. Even after 40 years of advancement in computing architecture (i.e. VAX, UNIX, and PC), 90% of commercial CAD packages in use today can still be traced to Adam [9]. Hanratty noted Adam succeeded where prior (and some subsequent) CAD systems failed. Rather than building Adam on a new language, Hanratty used an existing one (i.e. FORTRAN). The researcher's intention to write a tapered roller DFM computer program using the widely available java script language was based on Hanratty's work.

By the 1990's, less costly Windows operating systems had replaced UNIX workstation as the preferred CAD platform. Within the Window environment CAD packages were increasingly becoming Web enabled. Consistent with these computing trends, the researcher intended to use the Internet Explorer browser to interface with the java calculations.

Timken's desire to integrate DFM capabilities into its CAD system was consistent with recent developments in automated design critiquing systems [11]. At ISO, international standards for the exchange of product model data have been under development to codify design features. Gupta et al. proposed a computer interface module to critique CAD designs against these reference features. This research followed from Gupta et al. in that tapered roller designs were to be automatically critiqued. This research, however, went in a new computing direction. Rather than modelling features against standards, this work intended to apply experimentally derived equations to evaluate DFM.

3 Background

Timken's desire to integrate DFM capabilities into its CAD platform was consistent with its product differentiation strategy. To remain competitive in the price sensitive bearing industry, Timken was promoting upwards of 50% fuel economy for the most demanding bearing applications [12]. These applications typically required longer, steeper taper, and/or larger diameter rollers centreless ground to less than 2 μ m OOR. The researcher, upon review of Timken's production logs for one of its largest roller manufacturing facilities, noted when the product of roller length, taper, and nominal diameter exceeded 35 rads*mm^2, centreless grinding OOR scrap increased from an average of 4% to 7% of production (Figure 3-1).

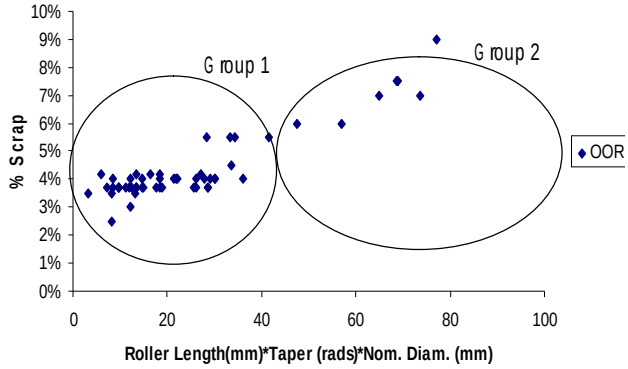
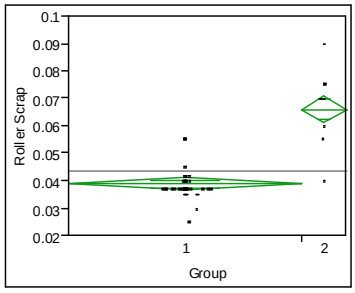


Figure 3-1 Rollers of Higher Geometry Factor Exhibited Higher OOR Scrap

Wilcoxon and Chi Squared testing (Figure 3-2) confirmed the difference in OOR scrap between the 2 roller geometry groups was statistically significant.



Wilcoxon / Kruskal-Wallis Tests (Rank Sums)

Level	Count	Score Sum	Score Mean	(Mean-Mean0)/Std0
1	47	1144.50	24.3511	-4.491
2	9	451.500	50.1667	4.491

1-way Test, ChiSquare Approximation

ChiSquare	DF	Prob>ChiSq
20.2696	1	<.0001*

Figure 3-2 Difference in OOR Scrap between Roller Geometry Factor Groups Was Statistically Significant.

This secondary data, however, was not collected under controlled testing conditions. Confounding factors and possible roller geometry interaction effects were unknown. Testing was required to determine whether correlations (or more specifically equations for computer modelling purposes) existed between roller geometry factors, OOR and maximum rev/sec.

4 Methodology

4.1 Test 1

Null hypothesis:

"Tapered roller length, included angle, and nominal diameter factors do not correlate with maximum roller rev/sec. during traverse centreless grinding"

Since the researcher expected interactions amongst the 3 roller geometry factors when determining maximum roller rev/sec, a 2^3 full factorial test space was used (Table 4-1).

Table 4-1 Roller Geometry Correlation with Maximum Roller Rev/Sec

	$8.2< D_{NR} <13mm$		$17.8> D_{NR} >13.1mm$	
	$2.8<\beta<4.4°$	$9° >\beta>4.$	$2.8<\beta<4.4°$	$9° >\beta>4.6°$
$12.9<R_L<22.6mm$	1	3	5	7
$35.7>R_L>22.7mm$	2	4	6	8

In each of the 8 treatments 2 different roller geometries were tested. For all roller geometries tolerances were the same. Samples of 5 sequential rollers were collected after 20, 40, and 60 minute grind intervals. If all 15 rollers were within roller tolerances, 60 min grind tests were successively repeated at 6 rollers / min higher productivity levels until at least 1 of the 15 sampled rollers was out of tolerance. This testing protocol was repeated two times for each of the 16 roller geometries in the test space. Maximum mean roller rev/sec values were recorded across each of the 8 factorial cells.

To correlate roller geometry factors with maximum rev/sec. required data exhibited independence of error, homogeneity of variance, normality, and signal effects at least 2 times natural variation [13]. To work within these data limitations, all maximum mean rev/sec values were natural log transformed. Transformation deflated variance, improved data normality, and maintained relative treatment effect size [14]. Since logarithm transformations tended to convert geometric relationships to linear ones [15], the researcher confined modelling to linear regressions.

4.2 Test 2

Null hypothesis:

"Roller length, taper, and nominal diameter factors do not correlate with OOR during traverse centreless grinding."

In Test 2, the sampled rollers from Test 1 were measured for OOR. The measurement involved a rotary table encoder linked to a computer data acquisition board. Over θ_i =0.088° equally spaced angles, radius

amplitudes were recorded. Per NIST Roundness Measurement standard 2.3.4.4.1 [16], the software calculated least squared centre (LSC) OOR. As was the case in determining maximum mean rev/sec, for each of the 8 roller geometry factorial combinations, the OOR values were natural log transformed.

5 Results

In combination roller length (R_L), nominal diameter (D_{NR}), and taper (β) were found to correlate with maximum roller rev/sec. At 96% coefficient of determination as nominal diameter, length, and taper increased maximum roller rev/sec decreased (Equation 5-1).

$$\text{Ln(rev/sec)} = 5.78 - 0.029R_L\ (\text{mm}) - 0.1116D_{NR}(\text{mm}) - 6.513\beta(\text{rads}) \quad (5\text{-}1)$$

When retesting using a different grinding process, a strong linear regression (at 97% coefficient of determination) also existed. However, slope and y-intercept terms were marked different from Equation (5-1).

The researcher concluded while the roller geometry 3-factor regression existed independent of grinding process, the slope and y-intercept values varied with it. Sources of the regression in both processes traced back to 3 grinding parameters (Fn, $\Delta\gamma$, and OOR).

Long rollers ground at higher maximum metal removal rate (i.e. MRR) compared to short rollers. As MRR increased grinding force normal (Fn) increased and maximum rev/sec decreased.

High taper rollers ground at higher pitch and yaw. At higher pitch and yaw changes in grinding angle (i.e. $\Delta\gamma$) along roller length increased and maximum rev/sec decreased.

As nominal diameter increased OOR increased. Rollers grinding at higher OOR ground at lower maximum rev/sec.

In regards to the roller geometry correlation with OOR, at 92% coefficient of determination, the researcher found as nominal diameter increased OOR increased (Equation 5-2).

$$\text{Ln(OOR}(\mu\text{m}))= -0.44 + 0.048D_{NR}(\text{mm}) \quad (5\text{-}2)$$

When repeating the test across a 2x larger sample group (with test controls intentionally removed), the researcher, again, found as nominal diameter increased OOR increased. At 69% coefficient of determination the regression was not weak. Moreover, the slope and y-intercept terms in the re-test closely matched the prior test.

It was possible to predict tapered roller OOR using nominal diameter. This was because synchronous vibrations of the regulating wheel (which spun the rollers during grinding) transferred at higher amplitude to larger nominal diameter rollers per revolution of the rollers. Equation (5-3) quantified how much of the regulating wheel sinusoidal amplitude transferred per roller revolution. Equation (5-4) projected this kinematic displacement normal to the grinding wheel per the nominal diameter set up angles. Using Equation 5-3 and 5-4, model fit predicting OOR improved to 87%.

$$A = ((\text{RW radial run out amplitude}) * \sin(\frac{D_{NR}(mm)}{RW_{ND}(mm)} * 360°)\cos\beta \quad (5\text{-}3)$$

$$A\frac{\cos(\theta+\phi_1)}{\cos(\theta-\phi_2)} \quad (5\text{-}4)$$

6 Conclusions

Based upon experimentally derived Equations 5-1 through 5-4, it was possible to quantify how tapered roller geometry determined such effects as manufacturing productivity and application energy efficiency. Using java script the researcher wrote a computer program to calculate these effects for 4 roller length, taper, and nominal diameter design combinations.

Since productivity equations changed depending on grinding process, a drop down menu allowed the user to select grinding process. If designed roller geometries were outside the test space, the software provided a warning that calculated values were being extrapolated (not interpolated). For each of the 4 design options the software calculated roller mass and productivity. From these values (and user defined inputs about material cost per kg and process costs/hr), manufacturing costs were calculated. When selecting a “Calc. & Plot” button, the user updated the productivity and cost bar chart for the 4 roller designs under review (Figure 6-1).

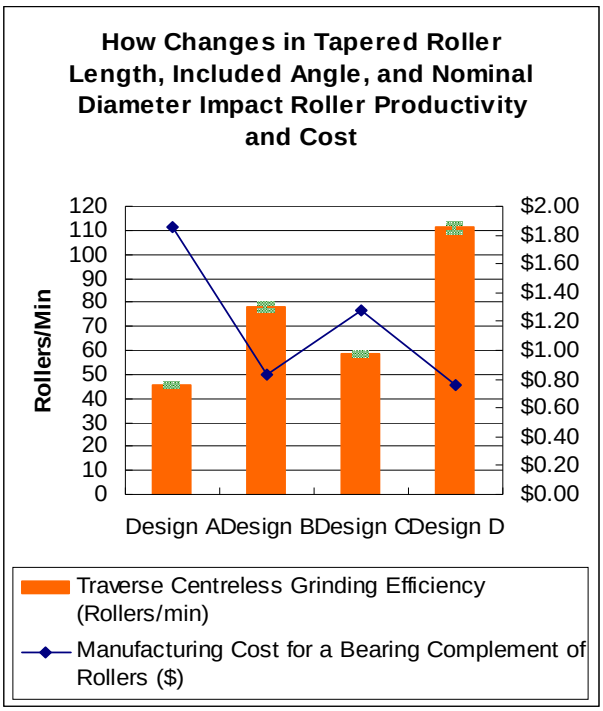


Figure 6-1 Computer Program Output for Productivity Design Review

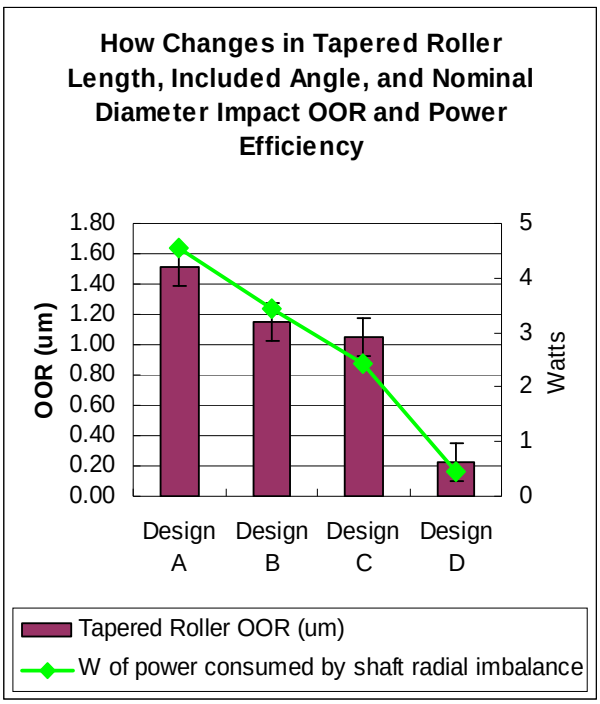


Figure 6-2 Computer Program Output for Energy Efficiency Design Review

In order to ensure roller design for optimised cost and productivity also met bearing application requirements, the DFM program calculated associated bearing load and speed ratings when "Calc & Plot" was selected.

Utilising Equations 5-3 and 5-4, the DFM program calculated roller OOR across the selected nominal diameter designs (relying upon user inputs about machine set-up angle and regulating wheel run out). With additional user defined inputs about application rotational speed and mass, the "Calc & Plot" button used Equations 1-1 and 1-2 to display OOR values and watts of power lost (in bar chart format per Figure 6-2).

7 Discussions

Per Gupta et al's proposal [11], DFM in CAD focused on automated comparisons made between reference features and the design under construction. Developments in DFM for CAD have, therefore, been measured in terms of maximizing reference features for real time guidance to the CAD user.

This research applied a different approach. Per CAD user defined inputs about process variables and costs, the software applied experimentally derived equations to critique the roller design. Categories considered were manufacturing productivity and application energy efficiency. Constraints the software ensured were satisfied were application load and speed rating.

For future CAD applications, the researcher concluded DFM equations could either be derived from (a) secondary work shop data or (b) experimental data. Method (b), while more costly and time consuming, was preferred. For example, it mitigated external factor affects on the data and enabled underlying causes to be determined. In either case, with data satisfying statistical test requirements, commercial software packages (i.e. Minitab, jmp, etc.) provided efficient means to determine the DFM equations. A disadvantage of such a quantitative DFM for CAD packages was the amount of process and cost data the user needed to provide the computer system. Users may not have had access to this type of data. In response to this criticism, the researcher

noted to perform successful DFM one needed to have detailed process and cost data.

8 Glossary

β = Full included angle of the tapered roller (°)

ϕ_1 = Grinding wheel angle of tangency with the roller periphery at the small end diameter (°)

ϕ_2 = Regulating wheel angle of tangency with the roller periphery at the small end diameter (°)

γ = Angle of roller tangencies with the grinding and regulating wheels (°)

θ = Support blade angle (°)

θ_i = Equally spaced angles per LSC radius measurements (radians)

ω = Angular velocity of the tapered roller bearing (rads/sec)

CADD = Computer Aided Drafting and Design

DFM = Design for Manufacturability

D_{NR} = Nominal roller diameter (mm)

F = Normal grinding force (N)

m= Load rotated on the bearing spindle (kgs)

Ln = natural log transformation

OOR = Out of Roundness (μ m)

P = Power required to rotate the load under imbalance (W)

R_L = Roller length (mm)

RW_{ND} = Regulating wheel nominal diameter (mm)

V_m = Linear speed of rotation for the load rotating on the bearings (m/sec)

9 References

[1] Botham, R. (2011) *Top 5 Advantages of CAD Design.* Available from: <http://ezinearticles.com/?Top-5-Advantages-of-CAD-Design-and-Drafting-Services-Outsourcing&id=3964502> [Retrieved Jan 2011]

[2] Crow, K (2001) Design For Manufacturability. Available from: < http://www.npd-solutions.com/dfm.html> [Retrieved Dec 2010]

[3] Barrenetxea, D. Alvarez, J., Marquinez, J. Avoid Principal Instabilities and Constraints in Through Feed Centreless Grinding. *Journal of Manufacturing Science and Engineering.* vol. 132

[4] Dall, A. (1946) Rounding Effect in Centerless Grinding. *Fall meeting of the Cincinnati Section of the American Society of Mechanical Engineers*. Cincinnati, USA: ASME.

[5] Snoeys, R. and Brown, D. (1969) Dominating Parameters in Grinding Wheel and Workpiece Regenerative Chatter. *Proceedings of the 10th international Machine Tool Design and Research Conference*, pg. 325-48

[6] *Calculating the Imbalance Force* (2009) Available from: <http://www.dliengineering.com/vibman/calculatingtheimbalanceforce.htm> [Retrieved 10 March 2009]

[7] Hahn, R., King, R.(1998) *Handbook of Modern Grinding Technology*. Princeton, MA: Service Network Press.

[8] *International Directory of Company Histories-Timken.* (2001). Available from: <http://www.fundinguniverse.com/company-histories/The-Timken-Company-History.html> [Retrieved 7 Feb 2010]

[9] *The CAD/CAM Hall of Fame (1998)* Available from: <http://www.americanmachinist.com/304/Issue/Article/False/9168/Issue> [Retrieved 10 Jan 2011]

[10] *CAD SOFTWARE* (2004) Available from:<http://www.cadazz.com/cad-software-history.htm> [Retrieved 12 Feb 2010]

[11] Gupta, S., Regli, W., Nau, D., (2010) Integrating DFM with CAD through Design Critiquing. *Concurrent Engineering: Research and Applications*. vol. 2, no. 2

[12] *P900 Bearings*. (2010). Available from: < http://www.timken.com/EN-US/PRODUCTS/BEARINGS/PRODUCTLIST/HIGHPERFORMANCE/pages/P900.aspx.> [Retrieved 10 Feb 2010]

[13] Keppel, G. (1973) *Design and Analysis: The Researcher's Handbook. Englewood Cliffs*, New Jersey: Prentice Hall Inc.

[14] Osborne, J (2002) *Notes on the Use of Transformations: Practical Assessment, Research, and Evaluation.* Available at: < http://pareonline.net/getvn.asp?v=8&n=6> [Retrieved 20 Feb 2010]

[15] Decision 411 Forecasting (2005). *The Logarithm Transformation.* Available from: <http://www.duke.edu/~rnau/411log.htm.> [Retrieved 20 Sept 2010]

[16] NIST/SEMATECH e-Handbook of Statistical Methods (2008) *Single Trace Roundness Design.* Available at: < http://www.itl.nist.gov/div898/handbook/mpc/section3/mpc3441.htm> [Retrieved 29 March 2009]

Biologically-inspired modelling and implementation of the human peripheral auditory system

Xin Yang, Mokhtar Nibouche and Tony Pipe

Bristol Robotic Laboratory, University of the West of England

Abstract

In this paper, a biologically-inspired acoustic signal processing system is developed in an FPGA (field programmable gate array) development board for sound source localisation. The hardware system is based on the modelling of the human peripheral auditory system, according to current developments in physiology and neuroscience. This novel auditory signal processing front-end transforms acoustic inputs to neural spike outputs in real-time for post-processing, just like the behaviour of the human auditory peripherals. Experiment results show the potential and ability of adapting this system in practical applications. The research provides a new solution for practical acoustic signal processing applications, and a uniform platform for both physiological research and acoustic engineering projects.

1. INTRODUCTION

The human auditory system deals with a wide range of everyday real life applications such as pitch detection, sound localisation and speech recognition, just to name few. It does the job extremely well, and far better than the current artificial acoustic mechanisms. If the mechanism of the auditory system could be modelled and reproduced, many applications would emerge, from hearing-aid devices to navigation and automatic recognition systems for robots. With the rapid development of silicon technology and electronic design automation (EDA) software, it is possible to model and implement a sophisticated auditory front-end in a single digital chip at a much lower cost than the traditional application specific integrated circuit (ASIC) design, but hundreds of times faster than the current computer software routines. Then the potential of building an "electronic ear" emerges.

1.1. Auditory System

The auditory system consists of the auditory periphery, as the front end sensing apparatus, and includes different other regions of the brain up to the auditory cortex. The human auditory periphery, as illustrated in Figure 1, consists of the outer, middle and inner ear. The inner ear, or the cochlea, is a coiled tube filled with fluid. The sound vibration is transmitted to the fluid, and then to the basilar membrane of the cochlea. The stiffness of the basilar membrane decreases exponentially along the length of the cochlea. This makes the basilar membrane act like a frequency analyser with the base part (near the oval window) responding to high frequencies, and the apical part (the far end) responding to lower frequencies. The sensory cells to detect the frequencies are the hair cells attached to the basilar membrane. There are three rows of outer hair cells (OHC) and one row of inner hair cells (IHC). For human, there are approximately 12,000 OHCs and 3,500 IHCs [11]. The movement of the OHCs and the basilar membrane is conveyed to the IHCs and causes a depolarisation, which in turn results in a receptor potential. Thus, IHCs release neurotransmitters, whose concentration change gives rise to nerve spikes into the auditory nerve.

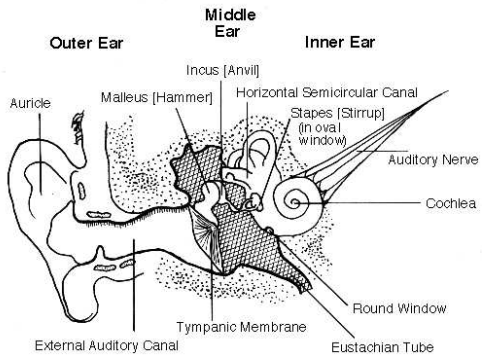


Figure 1. The human ear (taken from [11]).

1.2. Related Work

The auditory periphery 'transduces' acoustical data into train of spikes for more high level neuronal processing. Based on this, many researchers and engineers believe that modelling and implementation of an artificial auditory subsystems, especially the cochlea, will yield better performance in acoustic signal processing. The reported electronic cochlea implementations fall into two categories.

One is the task-oriented engineering approach, which treats the whole cochlea as filters in order to obtain the time/frequency information that can be used for differ-

ent kinds of post-processing. These are the majority of the reported implementations, including the first electronic cochlea proposed by Lyon and Meads [4], some recent FPGA implementations [6], [2], and [14].

The other category is the research-oriented signal processing approach, which analyses each stage of the biological acoustic signal processing in the auditory neurological system. There is less work in this category, and most of them focus on the hair cells and auditory nerve. Lim reported a pitch detection system [3] based on the Meddis' inner hair cell model [5], then Jones improved it to a four-stage pitch extraction system [1]. A spike-based sound localisation system was implemented by Ponca [8]. There was an analogue hair cell model implemented and then improved by van Schaik [13], [12].

The prototype implementation of the auditory subsystem in this paper belongs to the second category. It models a part of the signal processing of the human cochlea, by connecting two widely accepted models, the Patterson's Gammatone filter bank (GFB) [7] and the Meddis' inner hair cell (IHC) [5]. Compared to other existing work, the proposed hardware implementation is fully parameterised and highly scalable. It provides a good platform for further research, and can be developed at the front-end of an embedded auditory signal processing system.

2. PATTERSON'S GAMMATONE FILTER BANK

The GFB proposed by Patterson [7] is a set of parallel Gammatone filters, each of which responds to a specific frequency range. The Gammatone filter is a bandpass filter with gamma distribution well known in statistics. It describes the impulse response of a cat's cochlea, which is very similar to the human cochlea. The GFB provides a reasonable trade-off between accuracy in simulating the basilar membrane motion and computational load. Some improved models have been developed based on the original work, however, due to the increased hardware computation burden, the original model is adapted here. The impulse response of a Gammatone filter is:

$$h(t) = At^{N-1}e^{-2\pi bt}\cos(2\pi f_c t + \varphi) \qquad (t \geq 0, N \geq 1) \tag{1}$$

Where A is an arbitrary factor that is typically used to normalise the peak magnitude to unity; N is the filter order; b is a parameter that determines the duration of the impulse response and thus the filters bandwidth, f_c is the centre frequency, and φ is the phase of the tone.

Slaney developed a digital version of the GFB [9], then implemented it in his Matlab™ "Auditory Toolbox" [10]. Each digital Gammatone filter consists of four second-order sections (SOS) or Infinite Impulse Response (IIR) filters, as illustrated in Figure 2. The general Z transfer function for each IIR filter is:

$$H(z) = \frac{A_0 + A_1 z^{-1} + A_2 z^{-2}}{1 + B_1 z^{-1} + B_2 z^{-2}} \tag{2}$$

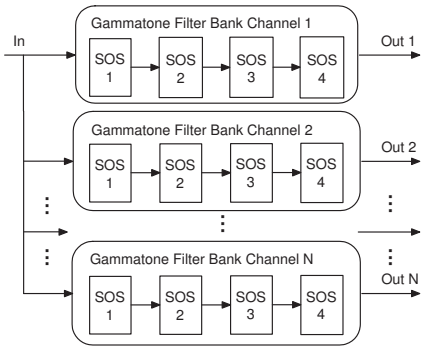


Figure 2. Slaney's digital Gammatone filter bank.

3. MEDDIS' INNER HAIR CELL MODEL

Meddis introduced the first hair cell model [5], which describes the transduction between IHCs and auditory nerves in a manner quite close to physiology by modelling both the short term and long term adaptation characteristics of the IHCs. Just like the case of the GFB model, the Meddis' IHC model has also been improved to adapt new findings in biology, however, for simplicity, the original model is chosen for this prototype implementation. The Meddis' IHC model can be described by a set of four nonlinear equations [5].

$$k(t) = \begin{cases} \frac{g(s(t)+A)}{s(t)+A+B} & \text{for } s(t)+A > 0 \\ 0 & \text{for } s(t)+A \leq 0 \end{cases} \tag{3}$$

$$\frac{dq}{dt} = y(1-q(t)) + rc(t) - k(t)q(t) \tag{4}$$

$$\frac{dc}{dt} = k(t)q(t) - lc(t) - rc(t) \tag{5}$$

$$P(e) = hc(t)dt \tag{6}$$

Where $k(t)$ is the permeability; $s(t)$ is the instantaneous amplitude; $q(t)$ is the cell transmitter level; $P(e)$ is the spike probability; A, B, g, y, l, x, r and m are constants based on statistics; and h is the proportionality factor, which can be set to different values.

The underlying structure of the model is illustrated in Figure 3 [5]. It is worth noting that there is also a Matlab™ implementation of the Meddis' IHC model in Slaney's "Auditory Toolbox" [10], which provides a reference for this design prototype.

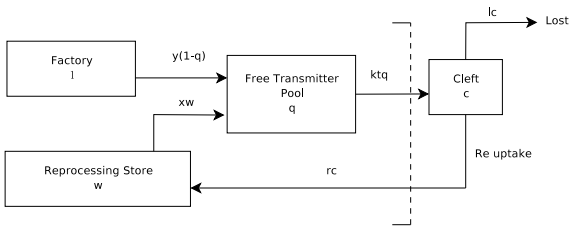


Figure 3. The Meddis' inner hair cell model [5].

4. SYSTEM IMPLEMENTATION

4.1. System Architecture

The proposed system architecture, as illustrated in Figure 4, consists of a GFB (a set of Gammatone filters) that mimics the behaviour of the basilar membrane, interfaced in a parallel fashion to the Meddis' IHC module through a bank of buffers. Each Gammatone filter is combined with a single Meddis' IHC to process for a particular range of frequencies (double-lined circle in Figure 4). The GFB (basilar membrane module) processes the incoming signal using parallel channels for different frequency ranges, and generates outputs that represent the vibration displacements along different parts of the biological basilar membrane. The Meddis' IHC module calculates the probability rate of neural spikes (spikes/second) corresponding to each output generated by the GFB module. A channel structure consisting of a Gammatone filter, a buffer and an IHC (double-lined circle in Figure 4) could be reconfigured to implement any of the system channels (1, 2, 3, . . . , n, . . . , N) through parameterisation. The system is also scalable, which means that an arbitrary number of channels can be generated, again through parameterisation. Simulation can be carried out either in software or hardware, however, for this prototyping stage; software simulation is preferred.

The specifications of the complete model are as follows. First, the "audible" frequency range of the basilar membrane module has to cover the human auditory range—from 200 Hz to 20 kHz [11]. A direct consequence is that the sampling frequency of the system sets to 44 kHz to allow a reasonable discrete representation. Second, perhaps a little arbitrarily, the number of channels is set to 20. Although the number of GFB channels should be the same as the number of IHCs in the cochlea (3,500), a compromise has to be made because of the constraints on hardware. Finally, the system must operate in real-time.

The general design methodology relies predominantly on IP-based blocks to model DSP primitives such as adders and multipliers using signed fixed-point bit serial arithmetic. Appropriate number representation, quantisation and overflow handling of the signals in the system are keys for a successful implementation. As such, the system input was coded as 14-bit signed fixed-point numbers, with 12 fractional bits. The output of the basilar membrane module was reduced to 10 bits with 8 fraction bits to relieve the computation load for the Meddis inner hair cell module. The output of the system was set as 14-bit with 12 fractional bits, in a similar fashion to the input.

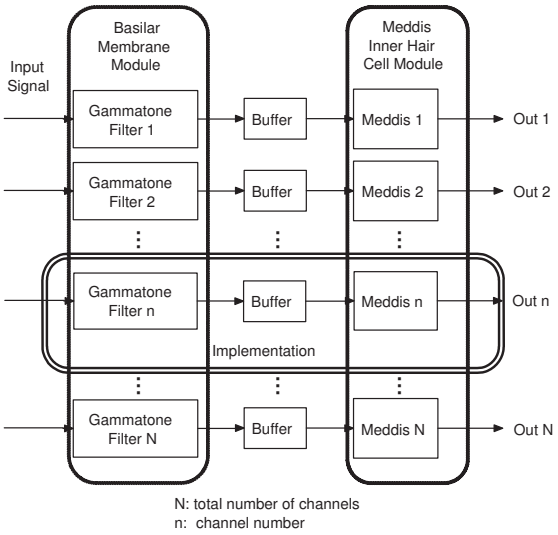


Figure 4. Structure for the whole model and the implemented channel.

4.2. Basilar Membrane Module

For the proposed design prototype, only one channel of the GFB was implemented using System Generator™ [16]. A channel or a Gammatone filter consists of four second-order IIR filters (SOS), each of which is implemented in the direct form structure, as illustrated in Figure 5. The calculations of the SOSs' coefficients are achieved by using Slaney's "Auditory Toolbox". There is no A_2 forward path as indicated in the transfer function of equation 2, simply because the coefficient A_2 is always zero for all the SOSs of any channel. The numerator coefficients (A_0 and A_1) of each SOS are scaled by $\sqrt[4]{\text{total gain}}$, resulting in the Gammatone filter to be scaled by the total gain which is calculated by the toolbox. This scaling narrows the dynamic range of the intermediate signals, and results in a reduced word length for the adders and multipliers. Table 1 presents an example of the calculated coefficients for the first channel of the GFB, where the total gain for this channel is 2.90294×10^{16}. Consequently, all the coefficients A_0 and A_1 in the table are scaled by 1.30529×10^4. It is worth noting that only the coefficient A_1 of the a SOS differ from those of the other 3 SOSs in the same channel, which makes a hardware optimisation possible for a single SOS.

Table 1. Coefficients of the IIR filters in channel 1.

IIR	A_0	A_1	B_1	B_2
SOS 1	0.29665	−0.10187	1.26532	0.56372
SOS 2	0.29665	0.47724	1.26532	0.56372
SOS 3	0.29665	0.13800	1.26532	0.56372
SOS 4	0.29665	0.23736	1.26532	0.56372

4.3. Meddis Inner Hair Cell Module

Slaney's Matlab™ implementation [10] of the Meddis' IHC model must be revised prior to be synthesised by AccelDSP™ [15]. This is not only because of the

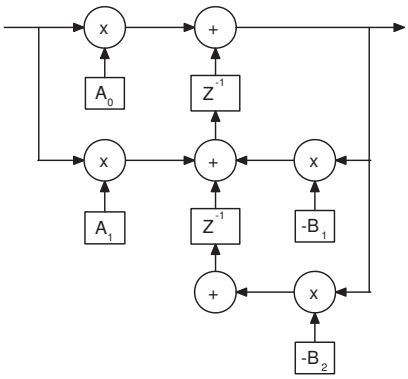


Figure 5. A second-order section of the Gammatone filter.

constraint requirements imposed by the AccelDSP^{TM} software, but also the demand for a real-time DSP system. The revised implementation processes the input signal in a fixed-point, bit-serial fashion, which leads to an unavoidable error compared to the original floating-point model, as illustrated in Figure 6. The investigation into the reported hardware implementation ([3],[1]) highlights this error as well. In reality, this is not a critical issue since the exact numerical values of the probability spike rate output are not essential for the spike generation (could be scaled up using the h coefficient in equation 6), however, the half-wave rectification, the saturation and the adaptation (both long term and short term) characteristics of the output are in return quite important [1]. The Meddis IHC module was generated based on this revised model using AccelDSP^{TM} . Figure 6 gives a comparison between the generated fixed-point model and the original floating-point model using a 1 kHz sine wave input.

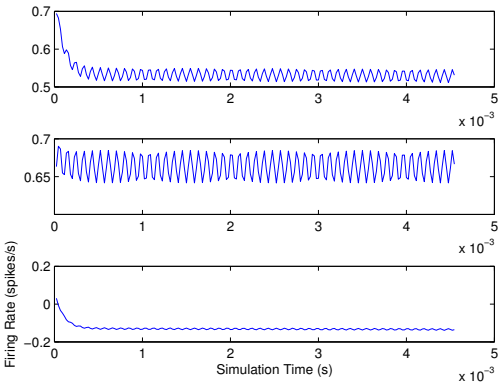


Figure 6. From top to bottom: outputs of the fixed-point Meddis' IHC model, the floating-point Meddis' IHC model and the error, with 1 kHz sine wave input.

4.4. Simulation and Synthesis Results

The test bench illustrated in Figure 7 was built to compare the simulation results of the original Matlab^{TM} floating-point model, the System Generator^{TM} fixed-point model and the FPGA hardware implementation model, however, the simulation here are only carried out in software at this prototyping stage. The input signal is generated by Simulink^{TM} blocks and can also be imported from files or even real-time external events. By an initialisation script, the total number of channels of the auditory subsystem can be set to an arbitrary number, in this case, 20, and the fixed-point model can be configured as anyone of the channels. The simulation results, shown in Figure 8, depicts the outputs of the 5_{th} channel under consideration for a step and then a sine wave input functions respectively. The hardware model generates closely matched outputs comparing with that of the software implementation in the simulation. The synthesis report illustrated in Table 2 indicates that the hardware utilisation is quite low (7%), except for the multipliers (32%), but the design can run in real-time. It also implies that only 3 channels can be implemented in parallel for this FPGA chip.

Table 2. Synthesis report of the first channel of the auditory subsystem.

Target Device	XC2VP30	
Synthesis Tool	XST v10.1.01	
Used Slices	993	7%
Used Slice Flip Flops	827	3%
Used 4 input LUTs	2,574	9%
Used RAMB16s	2	1%
Used MULT18X18s	44	32%
Max Frequency	17.505 MHz	

5. CONCLUSIONS AND FUTURE WORK

The paper presents the design and FPGA implementation of a bio-inspired hardware module that can be used as a front end apparatus in a variety of embedded auditory signal processing applications. The implementation consists of two sub-modules, Patterson's GFB and Meddis' IHC, linked together to mimic the behaviour of a single frequency channel of the auditory periphery. The proposed design is fully parameterised and highly scalable. The design prototype has been captured and then simulated using two integrated tools, System Generator^{TM} and AccelDSP^{TM} both from Xilinx^{TM} . The prototype works as expected and the design process is much faster than the traditional hardware description language (HDL) design flow. The resulting hardware structure was too large to accommodate a 20-channel parallel auditory subsystem; therefore, a time-shared multiplexing scheme is envisaged for future implementations. More optimisation can be achieved for the filter modules to improve the system performance and reduce the number of multipliers. The ultimate goal is to build a complete bio-inspired system that models the signal processing of the whole human auditory system.

References

[1] S. Jones, R. Meddis, S.C. Lim, and A.R. Temple. Toward a digital neuromorphic pitch extraction system. *Neural Networks, IEEE Transactions on*, 11 (4):978–987, July 2000.

[2] M. P. Leong, Craig T. Jin, and Philip H. W. Leong.

An fpga-based electronic cochlea. *EURASIP Journal on Applied Signal Processing*, 2003:629–638, 2003.

[3] S.C. Lim, A.R. Temple, S. Jones, and R. Meddis. Vhdl-based design of biologically inspired pitch detection system. In *Neural Networks,1997., International Conference on*, volume 2, pages 922–927, 9-12 June 1997.

[4] R.F. Lyon and C. Mead. An analog electronic cochlea. *Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing], IEEE Transactions on*, 36 (7):1119–1134, July 1988.

[5] R. Meddis. Simulation of mechanical to neural transduction in the auditory receptor. *Journal of the Acoustical Society of America*, 79 (3):702–711, 1986.

[6] A. Mishra and A.E. Hubbard. A cochlear filter implemented with a field-programmable gate array. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on [see also Circuits and Systems II: Express Briefs, IEEE Transactions on]*, 49 (1):54–60, Jan. 2002.

[7] R. D. Patterson, K. Robinson, J.W. Holdsworth, D. McKeown, C. Zhang, and M. Allerhand. Complex sounds and auditory images. In Y. Cazals, L. Demany, and K. Horner, editors, *Auditory Physiology and Perception*, page 429C446. Pergamon, Oxford, 1992.

[8] Marek Ponca and Carsten Schauer. Fpga implementation of a spike-based sound localization system. In *5th International Conference on Artificial Neural Networks and Genetic Algorithms - ICANNGA2001*, April 2001.

[9] Malcolm Slaney. An efficient implementation of the patterson-holdsworth auditory filter bank. Technical Report 35, Perception Group, Advanced Technology Group, Apple Computer, 1993.

[10] Malcolm Slaney. *Auditory Toolbox*, 1998.

[11] Barry Truax, editor. *HANDBOOK FOR ACOUSTIC ECOLOGY*. Cambridge Street Publishing, 2 edition, 1999.

[12] A. van Schaik. A small analog vlsi inner hair cell model. In *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on*, volume 1, pages 17–20, 25-28 May 2003.

[13] A. van Schaik and R Meddis. Analog very large-scale integrated (vlsi) implementation of a model of amplitude-modulation sensitivity in the auditory brainstem. *Journal of the Acoustical Society of America*, 105:811–821, 1999.

[14] C. K. Wong and Philip H. W. Leong. An fpga-based electronic cochlea with dual fixed-point arithmetic. In *Field Programmable Logic and Applications, 2006. FPL '06. International Conference on*, pages 1–6, 2006.

[15] Xilinx. *AccelDSP User Guide*. Xilinx, 10.1.1 edition, April 2008.

[16] Xilinx. *System Generator for DSP User Guide*. Xilinx, 10.1.1 edition, April 2008.

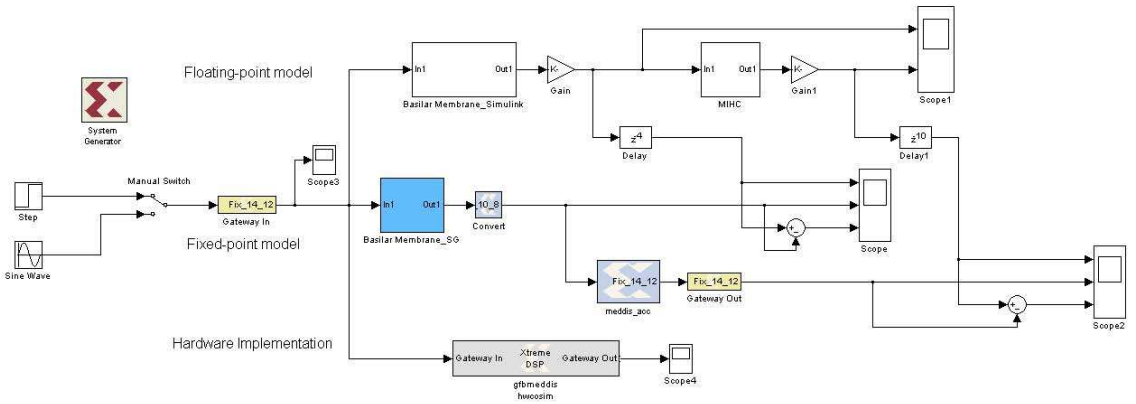


Figure 7. Test bench for the simulation of the prototype implementation of the auditory subsystem.

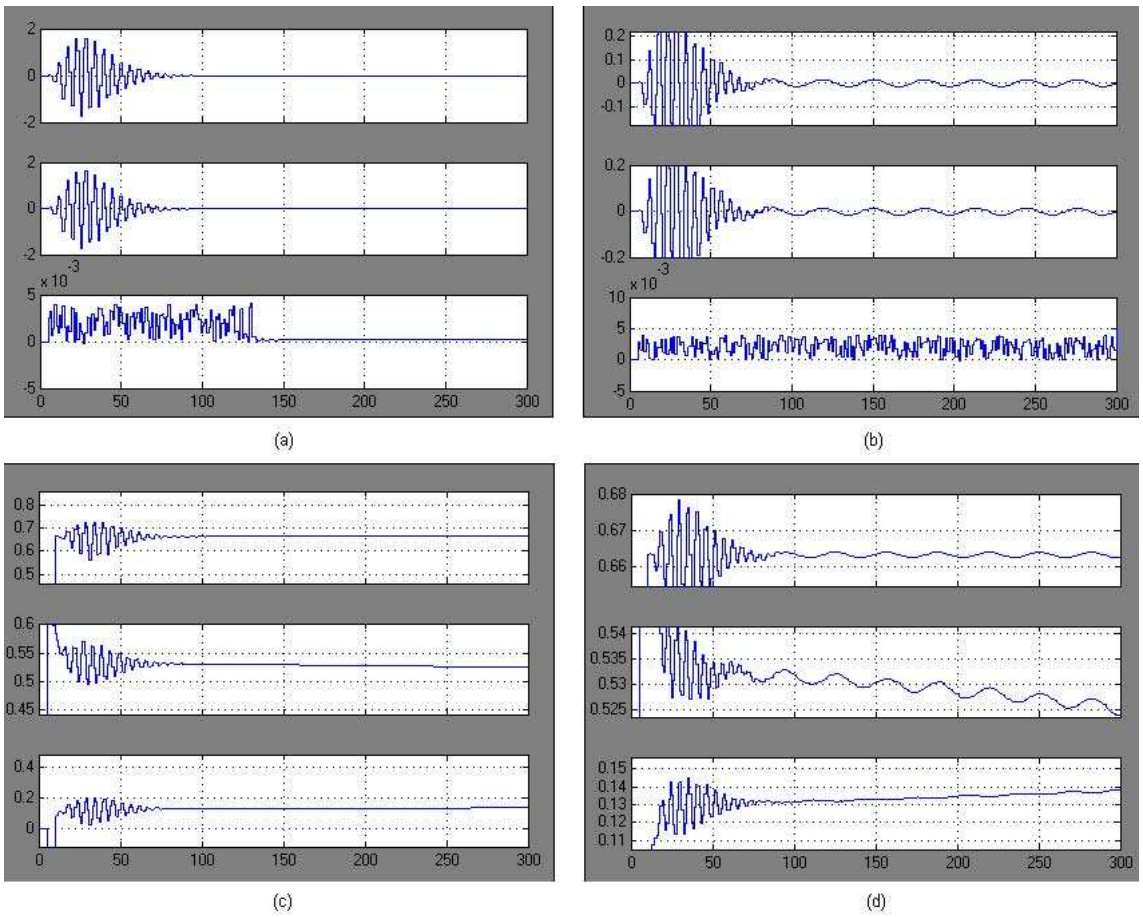


Figure 8. Simulation results for the 5_{th} channel. (a) the output of the basilar membrane module with step input; (b) the output of the basilar membrane module with 1 kHz sine input; (c) the output of the Meddis IHC module with step input; (d) the output of the Meddis IHC module with 1 kHz sine input. For each sub-figure, the output of the fixed-point hardware model is at the top, the output of the floating-point software module is in the middle, and the calculated error is at the bottom. The X-axis represents the simulation time, and the Y-axis represents the amplitude of the output.

Evolving sparse multi-resolution RVM classifiers

Andrew Clark and Richard Everson

Computer Science, University of Exeter

Abstract

The Relevance Vector Machine (RVM) provides a sparse probabilistic model for classification in which complexity is controlled using the Automatic Relevance Determination (ARD) prior. Unfortunately sparsity is also dependent on the kernels chosen and severe overfitting can occur when multi-resolution kernels are used.

In this paper we describe a multi-objective evolutionary algorithm (MOEA) which optimises RVMs in terms of true and false positive rates and model complexity, generating an approximate Pareto set of optimal trade-offs to select a classifier from. We describe a K-fold cross validation procedure for use during evolutionary optimisation in order to further regularise the generated RVMs and improve generalisation capability. Experiments run on a number of benchmark datasets using multi-resolution kernels demonstrate that the MOEAs are capable of locating markedly sparser RVMs than the standard whilst obtaining comparable accuracies.

1 Introduction

The Relevance Vector Machine (RVM) [20, 19] and its faster implementation, the fast RVM (fRVM) [21, 6], produces sparse probabilistic models for pattern recognition problems. Use of the kernel trick [1] allows models to be built in high-dimensional feature spaces at low computational cost with the advantage of a probabilistic formulation. Using the Automatic Relevance Determination (ARD) prior [10], outlined in section 2, the RVM 'switches off' basis functions for which there is little or no support in the data, producing a sparse representation. However, sparsity is not only controlled by the ARD prior, but also by choice of kernel [17] and severe overfitting occurs when multi-resolution kernels are employed. For regression problems this has been addressed by utilising a smoothness prior [17] but that methodology is not easily applicable to classification. In addition, it is often unclear what the misclassification costs are and, often, one wants to assess performance over a range of misclassification costs rather than a single one, typically through the use of the Receiver Operating Characteristic (ROC) curve.

To address these problems we propose a multi-objective optimisation method using an evolutionary algorithm in which we simultaneously optimise not only an RVM's true positive rate, T, and false positive rate, F, but also a new measure of the model complexity, C. By simultaneously optimising T, F and C we generate an approximation to the *Pareto Front* containing the best trade-offs between them. By controlling model complexity this way we reduce overfitting and so generate sparser models with equivalent generalisation performance. Like most training procedures, multi-objective evolutionary algorithms (MOEAs) are prone to overfitting on a training set and therefore we examine a K-fold cross validation scheme to control overfitting during the evolutionary optimisation process and examine the resulting effect on test error rates.

We begin by outlining the theoretical basis of the RVM and multi-objective optimisation, present our algorithm for multi-objective evolutionary optimisation of RVMs in section 3 and provide a comparison of the fRVM and MOEA. In section 4 we describe the K-fold cross validation scheme investigated and present results on a number of benchmark binary classification tasks in terms of accuracy and area under the ROC curve in section 5.

2 Relevance Vector Machines

The Relevance Vector Machine models binary classification using a logistic link function to map a linear combination of basis functions to a posterior class probability; thus if $\mathbf{x}$ is an input vector to be classified and $t \in \{0, 1\}$ is the corresponding target class label, then the posterior class probability is estimated as:

$$p(t \,|\, \mathbf{x}) = \frac{1}{1 + e^{-y(\mathbf{x})}} \tag{1}$$

where

$$y(\mathbf{x}) = w_0 + \sum_{m=1}^{M} w_m \phi_m(\mathbf{x}) \tag{2}$$

Here the w_m are the weights associated with the, usually nonlinear, basis functions $\{\phi_m\}_{m=1}^{M}$. For notational convenience we define $\phi_0 \triangleq 1$ and the $(M+1)$-dimensional vector of weights is denoted $\mathbf{w}$. Often the basis functions are derived from kernels, $K(\cdot, \cdot)$, centred at each of the observations $\{\mathbf{x}_n\}_{n=1}^{N}$ in a training set, so that $\phi_m(\mathbf{x}) = K(\mathbf{x}_m, \mathbf{x})$, however they may be quite general functions such as multi-resolution Gabor wavelets as used in image processing tasks [18]. In common with Support Vector

Machines, using basis functions defined via kernels permits inner product computations in the high-dimensional feature space spanned by the basis functions to be performed in the low-dimensional data space, affording great computational savings [1].

Sparsity in the RVM is achieved by placing an Automatic Relevance Determination (ARD) [10] prior

$$p(w_m \,|\, \alpha_m) \propto \alpha_m^{\frac{1}{2}} \exp\left(-\frac{1}{2}\alpha_m w_m\right) \tag{3}$$

over each of the weights, w_m, $m = 0, \ldots, M$. As [20] shows, this induces a heavy tailed Student-t prior over each of the weights w_m, favouring solutions in which probability mass is concentrated along the weight space coordinate axes. Consequently, sparse solutions in which weights are either 'switched on' (α_m small) or 'switched off' ($\alpha_m \to \infty$) are favoured over solutions with intermediate values of α_m.

Learning is achieved by type-II maximum likelihood estimation, in which the marginal likelihood

$$\mathcal{L}(\boldsymbol{\alpha}) = p(\mathbf{t} \,|\, \boldsymbol{\alpha}) = \int p(\mathbf{t} \,|\, \mathbf{w}) p(\mathbf{w} \,|\, \boldsymbol{\alpha}) d\mathbf{w} \tag{4}$$

is maximised with respect to the vector of hyperparameters $\boldsymbol{\alpha}$. In the regression case, with normally distributed observational noise of variance σ^2, the likelihood of a single observation is $p(t \,|\, \mathbf{w}, \sigma^2) = \mathcal{N}(t \,|\, y(\mathbf{x}), \sigma^2)$, so that (4) can be integrated analytically. Writing the training targets as a N-dimensional vector $\mathbf{t}$, we have

$$\log p(\mathbf{t} \,|\, \boldsymbol{\alpha}, \sigma^2) = -\frac{1}{2}\left[N \log 2\pi + \log |\mathbf{C}| + \mathbf{t}^T \mathbf{C}^{-1} \mathbf{t}\right] \tag{5}$$

with

$$\mathbf{C} = \sigma^2 \mathbf{I} + \boldsymbol{\Phi} \operatorname{diag}(\boldsymbol{\alpha})^{-1} \boldsymbol{\Phi}^T \tag{6}$$

and $\boldsymbol{\Phi}$ is the N by $(M + 1)$ design matrix: $[\boldsymbol{\Phi}]_{nm} = \phi_m(\mathbf{x}_n)$. The 'fast' RVM (fRVM) [6, 21] exploits a decomposition of the model covariance matrix $\mathbf{C}$ to yield an efficient greedy search procedure that at each step maximises the marginal likelihood with respect to a *single* α_m.

However, for classification problems (4) cannot be integrated exactly. Instead, a Laplace approximation [12] may be used to approximate the integrand around its mode, located using iterated reweighted least squares [11] (IRLS). This yields an expression for the log marginal likelihood similar to (5):

$$\log p(\mathbf{t} \,|\, \boldsymbol{\alpha}) \approx -\frac{1}{2}\left[N \log 2\pi + \log |\mathbf{C}| + \hat{\mathbf{t}}^T \mathbf{C}^{-1} \hat{\mathbf{t}}\right] \tag{7}$$

but with

$$\mathbf{C} = \mathbf{B} + \boldsymbol{\Phi} \operatorname{diag}(\boldsymbol{\alpha})^{-1} \boldsymbol{\Phi}^T \tag{8}$$

$$\hat{\mathbf{t}} = [\boldsymbol{\Phi}^T \operatorname{diag}(\boldsymbol{\alpha})^{-1} \boldsymbol{\Phi}^T + \mathbf{B}^{-1}](\mathbf{t} - \mathbf{y}) \tag{9}$$

Here $\mathbf{B} = \operatorname{diag}(y_n(1 - y_n))$, a matrix of heteroscedastic noise precisions necessary to map the classification problem to a regression one. Since the form of (8) is identical to (6), the fRVM scheme may be used to learn $\boldsymbol{\alpha}$ for classification.

2.1 Complexity measures for RVM classifiers

Unfortunately the sparsity control provided by the ARD scheme is not enough to prevent overfitting, a problem that can become particularly apparent when using highly resolving basis functions. In regression sparsity and overfitting can be controlled using a noise-dependent smoothness prior [17] of the form $p(\alpha_m \,|\, \sigma^2) \propto \exp\{-C(\boldsymbol{\alpha})\}$ where the degrees of freedom of the smoothing matrix $\sigma^{-2}\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^T$ of the solution is given by

$$C(\boldsymbol{\alpha}) = \sum_{m=0}^{M} \frac{1}{1 + \sigma^2 \alpha_m}. \tag{10}$$

Thus $C(\boldsymbol{\alpha})$ measures the number of active basis functions: those basis functions for which $\alpha_m < \sigma^{-2}$. However, the smoothness prior framework is difficult to incorporate into the RVM for classification. Whereas the degrees of freedom in the smoothing matrix (10) are calculated with respect to a single observational noise variance, the principal obstacle applying this to classification arises from the association of an effective noise precision $y_n(1-y_n)$ with *each* observation. Simple averages of the heteroscedastic noises are ineffective in properly controlling solution complexity and schemes that explicitly compute the trace of the (N by N) smoothing matrix are computationally expensive. The heteroscedastic noise appears in the design matrix rows but we wish to control the complexity of the solution by selecting from the columns i.e. the basis functions.

2.2 Measuring complexity of RVM classifiers

We adopt $C(\boldsymbol{\alpha})$ given by (10) as a measure of the RVM's complexity. As $\alpha_m \to \infty$ so the basis function ϕ_m is 'switched off' making no contribution to the complexity, whereas basis functions with small α_m contribute fully. Thus $C(\boldsymbol{\alpha})$ measures the magnitude of the activation of the basis functions.

Although σ^2 is not well determined for classification problems, $C(\boldsymbol{\alpha})$ gives an ordering for the complexity of solutions independent of the particular value of σ^2. We therefore propose minimising $C(\boldsymbol{\alpha}, \sigma^2)$ for fixed $\sigma^2 = 1$ while simultaneously maximising measures of the solution accuracy.

As noted, straightforward greedy optimisation schemes similar to the fRVM are not available using this complexity measure and we therefore turn to evolutionary optimisation, which opens up the possibility of simultaneously optimising the true and false positive rates and this measure of model complexity.

3 Evolving RVM Classifiers

The performance of a binary classifier when the costs of misclassification are unknown may be summarised using the Receiver Operating Characteristic (ROC) curve [13]. Briefly, one focuses on one class, the *positive*, $t = 1$, class, and $\mathbf{x}$ is assigned to the positive class if $p(t \,|\, \mathbf{x}) > \lambda$ for some threshold λ. The ROC curve is generated by plotting the true positive rate T versus the false positive rate

F whist λ is varied from 0 to 1, corresponding to varying the ratio of misclassification costs from infinitely more costly to mis-assign to the negative class to infinitely more costly to mis-assign to the positive class. Determination of the best ROC over all possible parametrisations may be regarded as a two-objective optimisation problem where the goals are to simultaneously maximise the true positive rate and minimise the false positive rate. Here we generalise this, seeking to optimise both these rates over RVMs with differing numbers of active basis functions and regarding each RVM's complexity, as measured using the method discussed in section 2.2, as a third objective to be minimised.

3.1 Multi-objective minimisation and Pareto optimality

When performing multi-objective optimisation one seeks to maximise or minimise D objectives, which are functions, $f_i(\boldsymbol{\theta})$, $i = 1, \ldots, D$ of P parameters or decision variables, $\boldsymbol{\theta} = (\theta_1, \theta_2, , \ldots, , \theta_P) \in \boldsymbol{\Theta}$. It can be assumed, without loss of generality, that the objectives are to be minimised, so that the problem may be expressed as follows:

$$\text{Minimise} \quad \mathbf{f}(\boldsymbol{\theta}) \equiv (f_1(\boldsymbol{\theta}), \ldots, f_D(\boldsymbol{\theta})) \qquad (11)$$

noting that $\boldsymbol{\theta}$ may be subject to additional constraints.

Dominance is generally used to compare two multi-objective quantities $\mathbf{f}$ and $\mathbf{g}$. If $\mathbf{f}$ is no worse than $\mathbf{g}$ for all objectives and wholly better on at least one objective then $\mathbf{f}$ *dominates* $\mathbf{g}$ or $\mathbf{f} \prec \mathbf{g}$. Thus $\mathbf{f} \prec \mathbf{g}$ iff:

$$f_i \leq g_i \forall_i = 1, \ldots, D \text{ and } f_i < g_i \text{ for at least one } i \quad (12)$$

Domination in the objective space can extended to parameter space: thus if $\mathbf{f}(\mathbf{a}) \prec \mathbf{f}(\mathbf{b})$ it is said that $\mathbf{a}$ dominates $\mathbf{b}$: $\mathbf{a} \prec \mathbf{b}$. The dominates relation is not a total order and therefore two solutions are deemed *mutually non-dominating* if neither dominates the other. A set E is said to be a *non-dominating* set if no element of E dominates any other element in it:

$$\mathbf{a} \not\prec \mathbf{b} \quad \forall \mathbf{a}, \mathbf{b} \in E \qquad (13)$$

With a slight abuse of notation we write $E \prec \mathbf{a}$ if there is at least one member of the set E that dominates $\mathbf{a}$.

A solution is considered *Pareto optimal* if no other feasible solution dominates it. The set of all Pareto-optimal solutions is referred to as the *Pareto set* and the image in objective space under $\mathbf{f}$ of the Pareto set is known as the *Pareto-optimal front*, $\mathcal{P}$. The Pareto set thus represents all possible optimal trade-offs between competing objectives.

When performing binary classification, the subject of this paper, often the goal is to maximise the true positive rate T, whilst simultaneously minimising the false positive rate F. Here, we write a particular model as $\boldsymbol{\theta}$ where $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \lambda)$, a set of hyperparameters, $\boldsymbol{\alpha}$, and a classification threshold, λ. We then try to find the Pareto set of $\boldsymbol{\alpha}$ and λ combinations that maximise the true positive rate while minimising the false positive rate and complexity.

Algorithm 1 Multi-Objective Evolutionary Algorithm

Require: $\mathcal{D}$ *Data feature-target pairs*
$\boldsymbol{\alpha}$ *Initial solution*

1: $E := \texttt{initialise}(\mathcal{D}, \boldsymbol{\alpha})$
2: `while not converged`
3: $\quad \boldsymbol{\alpha} := \texttt{select}(E)$
4: $\quad \boldsymbol{\alpha}' := \texttt{perturb}(\boldsymbol{\alpha})$
5: $\quad$ `for` $\lambda' := 0 : \delta : 1$
6: $\quad\quad F(\boldsymbol{\alpha}', \lambda'), T(\boldsymbol{\alpha}', \lambda'), C(\boldsymbol{\alpha}') := \texttt{evaluate}(\boldsymbol{\alpha}', \lambda', \mathcal{D})$
7: $\quad\quad \boldsymbol{\theta}' := \{\boldsymbol{\alpha}', \lambda'\}$
8: $\quad\quad E := \texttt{nondom}(E \cup \{\boldsymbol{\theta}'\})$
9: $\quad$ `end`
10: `end`

3.2 Multi-Objective Optimisation of RVMs

In our algorithm we seek to find the Pareto set in terms of $(T(\boldsymbol{\theta}), F(\boldsymbol{\theta}), C(\boldsymbol{\theta}))$ using multi-objective optimisation. Multi-objective evolutionary algorithms have been used for ROC optimisation in medical and safety related systems [2, 8, 15]. The algorithm performs a greedy search using the dominates relation, based on T, F and C, to compare solutions, whilst maintaining an elite archive, E, of non-dominated solutions approximating the Pareto set. On each iteration of the optimisation a member of E, $\boldsymbol{\theta}$, is selected and perturbed to generate a new solution $\boldsymbol{\theta}'$. If $\boldsymbol{\theta}'$ is not dominated by existing members of E it is added to E and any newly dominated elements of E are eliminated. As a result, the archive can only approach the true Pareto front for the training data. Details of the procedure are given in algorithm 1.

For numerical convenience all α_m are restricted to a finite range: $\alpha_{\text{min}} \leq \alpha \leq \alpha_{\text{max}}$. If an α_m should exceed α_{max} the basis function is deemed to be 'switched off'. Conversely, α_m values less than α_{min} are set to α_{min}. In the experiments reported here we chose $\alpha_{\text{min}} = 10^{-12}$ and $\alpha_{\text{max}} = 10^{12}$, but results are not sensitive to the particular thresholds used.

The algorithm begins (line 1) by initialising the archive with a provided $\boldsymbol{\alpha}$; this may be a randomly selected vector, or a $\boldsymbol{\alpha}$ found using an fRVM trained on the training features $\mathbf{x}$ and labels $\mathbf{t}$ which together we denote by $\mathcal{D}$. Here we use an $\boldsymbol{\alpha}$ with one randomly chosen $\alpha_m = \alpha_{\text{min}}$ and all other α_n switched off, $\alpha_n = \alpha_{\text{max}}$ for $n \neq m$. This solution is then evaluated for $0 \leq \lambda \leq 1$ in steps of size δ to create an initial mutually non-dominating set E which is the ROC curve for $\boldsymbol{\alpha}$.

On each iteration an $\boldsymbol{\alpha}$ is selected from the set of unique $\boldsymbol{\alpha}$s within E and perturbed to form $\boldsymbol{\alpha}'$ (lines 3 & 4). By selecting from only the unique $\boldsymbol{\alpha}$s in E we avoid bias towards any particular combination of α values (there may be more than one solution with the same $\boldsymbol{\alpha}$ but different λ). Between 1 and 3 components of the selected $\boldsymbol{\alpha}$ are perturbed in one of the following ways chosen with equal probability: (i) an α_m corresponding to an active basis function is selected at random from the active components and α_m perturbed as:

$$\log_{10} \alpha'_m = \log_{10} \alpha_m + \epsilon \qquad (14)$$

where ϵ is drawn from a heavy-tailed Laplacian distribu-

tion, $p(\epsilon) \propto e^{-|\epsilon|/2}$, encouraging exploration [22]; (ii) a randomly selected active basis function is switched off by setting α_m to ∞; (iii) a randomly selected inactive basis function is switched on by setting α_m to a number uniformly drawn in logarithmic space between α_{min} and α_{max}.

IRLS is used to train weights $\mathbf{w}$ based on $\boldsymbol{\alpha}'$ allowing the true positive rate, T, false positive rate, F, and complexity, C, of the resulting model to be evaluated over a range of thresholds, λ' (lines 5 & 6). Each $(\boldsymbol{\alpha}', \lambda')$ pair forms a new perturbed parametrisation $\boldsymbol{\theta}'$ which, if it is not dominated by another parametrisation already in E, is added to E and any elements of E that are dominated by it removed. These operations are represented in the algorithm by the procedure $\texttt{nondom}(A)$ which returns the mutually non-dominating elements of the set A.

The algorithm is particularly efficient because for each perturbed parametrisation, $\boldsymbol{\alpha}'$, a large number of solutions of the same complexity may be cheaply generated by varying the threshold λ'. The algorithm is able to make simultaneous perturbations of more than one α_m each iteration, helping avoid local maxima encountered by the fRVM which perturbs only a single α_m at each step and may therefore have difficulty exchanging a solution for a better one if the intermediate step is worse. When no new additions have been made to E for 20 iterations we attempt to force more diversity in the front by deliberately perturbing α_m values by turning them on or off.

3.3 Illustration

We provide an initial illustration of this method by training the fRVM and our MOEA on a subset of the Banana dataset [14]. The Banana dataset is 2-dimensional, allowing visualisation of results, and consists of 400 training and 4900 test points for each provided fold with an overall positive class fraction of 44.7%. This presents a dataset in which the training observations are dense making it relatively easy to approach the Bayes error rate and so, in order to provide a more challenging problem and demonstrate the efficacy of the MOEA, for each of the first 10 folds provided we randomly select a subset of 100 points, maintaining the class balance, with which to train the fRVM and MOEA. Gaussian kernels were centred on each of the training points: $\phi_m(\mathbf{x}) \propto \exp\{-\|\mathbf{x}-\mathbf{x}_m\|^2/r^2\}$. We used kernels of widths $r = r^*, r^*/2$ and $r^*/4$ where $r^* = 0.25$, chosen to present the case where a highly resolving set of kernels have been selected.

In Figure 1 the Pareto front generated by the MOEA for the first fold of the dataset is shown, containing 798 solutions with true and false positive rates ranging from 0 to 1 and complexities ranging from 1.41×10^{-12} to 18.16.

Although the MOEA provides an entire Pareto front of solutions with a range of true and false positive rates, it is useful to select a single solution from the front for comparison. The bottom panel of Figure 1 shows the decision boundary of the solution with the highest accuracy (94%) on the training data and its location on the front ($T = 0.9831$, $F = 0.1220$, $C = 5.9957$) is marked in the top panel of Figure 1.

As the figure shows, the fRVM and MOEA solutions produce somewhat different decision boundaries. While

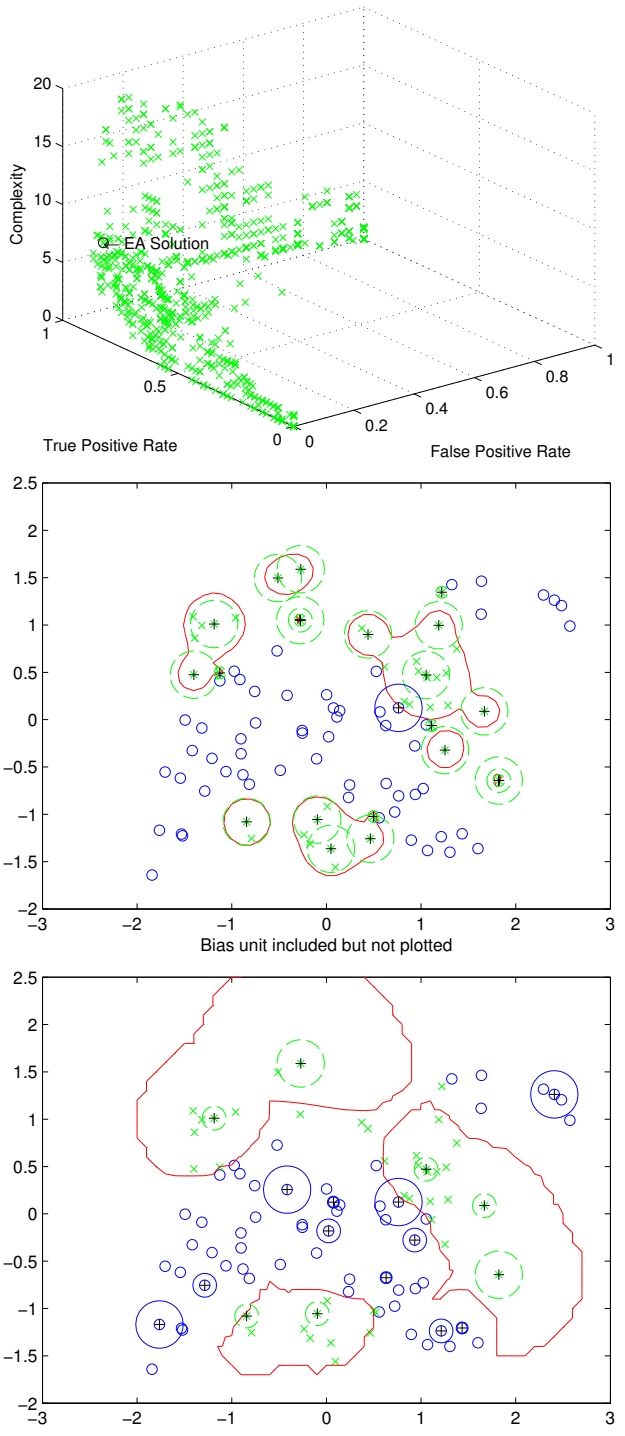


Figure 1: Top: Approximate Pareto front of solutions trained by the EA algorithm for Banana data fold 1 subset. Middle: Decision boundary for the fRVM solution. Bottom: EA solution with the highest accuracy on the training data. Positive class training data points are represented by the crosses and negative class by circles. Selected relevance vectors marked with +s and circles surrounding them represent the width of the selected kernel. Decision boundaries are shown by the red line

the fRVM has utilised more relevance vectors (active basis functions) than the MOEA solution (25 compared to 18) it has clearly overfit the training data, giving a decision boundary tightly wrapped around the positive class

members, only achieving a 68.08% test rate despite a 99% training rate. Conversely, the sparser EA solution has generalised comparatively well, selecting basis functions centred on observations from both classes and giving a wider decision boundary, achieving training and test rates of 94% and 84.63% respectively.

Over the ten folds the fRVM has an average training accuracy of 98.80% and test accuracy of 77.38% using 28.40 relevance vectors on average. The most accurate solutions for training data selected from the fronts generated by the MOEA achieve average training and test accuracies of 94.50% and 80.93% respectively with an average 21.00 relevance vectors, producing slightly sparser solutions than the fRVM.

The training and test accuracies discrepancy for the fRVM indicates that it has overfit the data, and therefore generalises poorly, a phenomenon also noticed for regression problems using multiple-scale kernels with the RVM [17]. It can be seen, therefore, that even on this small example the fRVM overfits whereas the MOEA incorporates some regularisation by excluding complex solutions from the archive which do not have better true and false positive rates than an existing member of the archive.

Although, with single-width kernels, regularisation can be achieved by "tuning" the kernel widths, such a strategy is not possible with multiple-resolution kernels. The MOEA provides a range of solutions at different complexities, however the method of selecting the solution with maximum *training* accuracy clearly also suffers from overfitting. Indeed, most solutions located by the MOEA with sufficiently high complexities will have overfit to the training data. Although $C(\boldsymbol{\alpha})$ measures the solution complexity it is unclear how an appropriate $C(\boldsymbol{\alpha})$ should be chosen *a priori*. In the following section we therefore examine a K-fold cross validation method for controlling overfitting during the evolutionary optimisation process.

4 Cross Validation for EAs

In machine learning, a common approach to controlling complexity is to penalise the likelihood (or error on the training data) with a regularisation term that increases as the solution becomes more complex. The degree of regularisation is then adjusted via a hyperparameter whose value must be determined, usually by cross validation on a reserved portion of the data (see [4] for a review of cross validation and alternative methods). The Bayesian point of view avoids this problem to a certain extent by setting priors for the problem parameters and averaging over posterior distributions, but maximum *a posteriori* learning may be regarded as a penalised likelihood method and careful setting of the prior (hyper-)parameters may be necessary to achieve good generalisation.

It would be computationally expensive to make multiple runs of the evolutionary optimiser to determine the best value of a regularising hyper-parameter and therefore we introduce a scheme in which we maintain an archive of solutions where we evaluate an average true and false positive rate based on a K-fold splitting of the training data.

K-fold cross validation is commonly used in statistical pattern recognition to ascertain the best regularisation parameter [5]. In this, the available training data is randomly partitioned into K disjoint folds and each of the K folds used as a surrogate test set, a *validation set,* on which to estimate the generalisation error of a model trained on the remaining $K-1$ folds. Suitable regularisation parameters are then those that yield the lowest validation error. The variance in the validation error estimate is reduced by averaging over the K validation sets; often 5 or 10 folds is sufficient [4].

Algorithm 2 K-Fold MOEA

```
1:  {D^(k)} := split(D)     k = 1,...,K
2:  E := initialise({D^(k)}, α)
3:  while not converged
4:    α := select(E)
5:    α' := perturb(α)
6:    C(α') := complexity(α')
7:    for λ' := 0 : δ : 1
8:      for k = 1,...,K
9:        w := IRLS(α', D_\k)
10:       F_k(α', λ'), T_k(α', λ') :=
                      evaluate(α', λ', w|D_k)
11:     end
12:     F(α', λ') := (1/K) Σ_{k=1} F_k
13:     T(α', λ') := (1/K) Σ_{k=1} T_k
14:     θ' := {α', λ'}
15:     E := nondom(E ∪ {θ'})
16:   end
17: end
```

The scheme presented here treats the ARD parameters α_m as regularisation hyperparameters and looks for those that give the best generalisation performance. Solutions are only accepted into the archive if they are not dominated by any elements of the archive, as measured by the average over K validation sets. Algorithm 2 is a straightforward modification of algorithm 1.

When partitioning the data into the K folds we consider the potential problems that may result as it is possible that, particularly for small datasets, partitioning may result in splits that are quite dissimilar. For example, if one were splitting the data into 2 folds, if the data consists of two clusters a poor split might assign most of one cluster to one partition and most of the other cluster to the other. As a result, a poor partition can result in a large reduction in the size of the archive, which inhibits convergence, and lead to the learning of non-representative models.

In order to address this we use the "nearly homogenous partitioning" method proposed in [3] to split $\mathcal{D}$ into K folds. Starting from a randomly chosen datum, homogeneous partitioning traces a path through the data from nearest neighbour to nearest neighbour; every k^{th} point on this path is assigned to the k^{th} fold. This yields K statistically similar folds on which to train and validate. As recommended by [3], the partitioning procedure is initialised by selecting a point from $\mathcal{D}$ at random and using the *furthest* point from that as the starting point for the partitioning algorithm.

The algorithm initialises (line 1) K "nearly homogenous" folds $\mathcal{D}^{(k)}$ and the average performance of an initial solution $\boldsymbol{\alpha}$ is evaluated on these folds to initialise the

archive E (line 2). As before an $\boldsymbol{\alpha}$ selected from the archive E is perturbed to form $\boldsymbol{\alpha}'$ (lines 4 – 5). At lines 8 – 11 the algorithm loops through the K folds, using IRLS to train weight vectors $\mathbf{w}$ based on $\boldsymbol{\alpha}'$ and data from all but the k^{th} fold; true and false positive rates are then evaluated on the k^{th} fold. These rates are *averaged* over the folds (lines 12 – 13) and if the solution is not dominated, in terms of these average T and F as well as the complexity C, by any solution in the elite archive E then it is added to E. Any solutions in E dominated by $\boldsymbol{\alpha}'$ are eliminated.

5 Results

Here, results on standard benchmark datasets comparing the MOEAs with the standard RVM trained using the fRVM algorithm are provided. As argued, the efficacy of the RVM depends on the regularisation provided by the kernels and for the fRVM we therefore adjusted the kernel widths to match the sparsity and test error rates reported by [20]. Denoting this tuned kernel width by r^*, we then constructed over-complete dictionaries comprising kernels of width r^*, $r^*/2$ and $r^*/4$. Using this expanded set of kernels, RVMs were trained using the fRVM method and the MOEA and its K-fold variant were used to find the approximate true positive vs false positive vs complexity Pareto front. For the K-fold cross validation scheme we present results for 2, 5 and 10 folds.

In order to compare with the results in [20], we employ the datasets assembled in [14], details of which can be found in table 1. Note, however, that the results in [20] were obtained using a gradient ascent method to maximise the approximate marginal likelihood.

In order to assess performance under situations of more extreme class imbalance we make use of the MNIST numerals dataset [9], looking to distinguish images of the numeral 5 from images of the other numerals and use only 1000 of the 60000 training samples, down-sampled from 28×28 pixels to 13×13 pixels, drawn in equal proportions from each of the underlying digit classes. As with the other datasets use a Gaussian kernel.

Even though the MOEA schemes provide a range of solutions with varying true and false positive rates across the front, in order to make a direct comparison with the fRVM a single solution must be selected. Therefore, the EA solution from the estimated Pareto front with the highest accuracy over *all* the training data in that fold was selected, i.e., where training data has been split for cross validation we recombine it and the archives and select the solution that performs best overall on the training data. Results for the MOEA schemes and fRVM are presented in table 2. We report average accuracies, along with the standard deviation over 10 splits of the data [14].[1]

For the Banana subset, Breast Cancer, Titanic, German and Pima datasets all the EA methods have test accuracies that are indistinguishable from those of the fRVM at the 5% significance level using a Mann-Whitney-Wilcoxon test. For the full Banana data the 5 fold scheme results are significantly different from the fRVM though the rest are not significantly different. The fRVM achieves significantly better accuracies than the MOEAs on the Waveform data.

The MOEA methods have thus located solutions of comparable accuracy to the fRVM but with markedly lower complexity and we draw particular attention to the results for the Titanic and German datasets where solutions achieving comparable accuracy to the fRVM, but with far fewer relevance vectors, have been located by our EAs.

The training and test accuracies for the basic MOEA are generally lower than the accuracies achieved by the fRVM alone, and the discrepancies between MOEA test and training rates are smaller than those for the fRVM, which frequently appears to overfit on training data. Some regularisation of the RVM is provided by the basic EA because a solution may not enter the elite archive if another is in the archive with identical or better T and F and lower complexity, thus ensuring only low complexity solutions enter the archive. Additional regularisation is provided by the K-fold scheme, which is evident from the smaller numbers of relevance vectors found.

As well as finding single solutions with high accuracy, the multi-objective schemes presented allow the location of a range of solutions spanning the full range of true and false positive rates. In order to compare these we calculate the area dominated by the Pareto front in the false positive – true positive plane. These areas, presented in table 3, are analogous to the area under the ROC curve and reflect the classifier's ability to separate the classes. Pareto fronts for the evolutionary optimiser are projected onto the false positive – true positive plane by ignoring the complexity and then finding the non-dominated set on the *training* data; the area dominated by this set on the *test* data is reported in the column labelled 'Test'. Area is calculated as the area strictly dominated by the front, that is, the area dominated by the attainment surface [23, 7]. As the table shows, the evolutionary algorithms locate a range of solutions that perform as well as, if not better than, the fRVM. 5 or 10 folds is clearly sufficient for adequate regularisation on these datasets.

6 Conclusion

We have presented a multi-objective evolutionary algorithm for training relevance vector machines for classification tasks, which allows solutions over the full range of true and false positive rates to be generated. When working with multi-resolution kernels, such as wavelets, the RVM tends to overfit to the training data and thus generalise poorly. By including a complexity measure as an objective to be minimised a degree of regularisation is provided by filtering out complex solutions that do not have superior true and false positive rates to those already in the elite archive.

A K-fold cross validation scheme to prevent overfitting has been examined which, in conjunction with the evolutionary algorithm, locates solutions with far fewer relevance vectors than the fRVM, but with comparable accuracies. Preventing overfitting is a common problem in

[1] The popular Pima data, available at `http://www.stats.ox.ac.uk/pub/PRNN`, is not included in [14]. We report results on Ripley's split [16] and for 9 random splits thereof where the number of training and test examples were of equal size to those in Ripley's split.

Data Set	**No. Training**	**No. Testing**	**Dimensions**	**Positive class fraction (%)**
Pima	200	332	7	33.27
Banana	400	4900	2	44.70
Banana Subset	100	4900	2	44.70
Breast Cancer	200	77	9	30.65
Titanic	150	2051	3	32.30
Waveform	400	4600	21	32.95
German	700	300	20	29.93
MNIST: 5 v All	1000	10000	169	9.04

Table 1: Details of benchmark datasets. Note that for the MNIST data we only use 1000 training points and down-sample the data so the number of features is 169.

	Banana			Banana Subset —		
Classifier	Train (%)	Test (%)	RVs	Train (%)	Test (%)	RVs
fRVM	94.05 (1.09)	**88.59 (0.75)**	29.70 (4.14)	98.80 (1.87)	77.38 (6.72)	28.40 (7.06)
MOEA	93.28 (1.27)	87.83 (0.87)	25.90 (5.57)	94.50 (1.78)	80.93 (2.20)	21.00 (5.87)
2 fold	92.48 (1.03)	88.09 (1.11)	25.30 (6.25)	93.60 (2.37)	**82.37 (2.31)**	17.20 (4.54)
5 fold	92.73 (1.19)	87.26 (0.68)	27.20 (4.16)	93.60 (2.50)	81.31 (1.72)	16.20 (6.39)
10 fold	92.97 (1.08)	88.19 (0.99)	24.70 (7.80)	92.70 (1.49)	80.29 (1.58)	16.80 (6.12)
	Breast Cancer			Titanic		
Classifier	Train (%)	Test (%)	RVs	Train (%)	Test (%)	RVs
fRVM	93.25 (1.34)	69.35 (4.42)	36.30 (3.83)	79.80 (2.95)	**77.71 (1.38)**	69.90 (22.12)
MOEA	83.45 (1.52)	68.83 (5.61)	19.90 (8.23)	80.00 (2.85)	77.54 (1.37)	7.90 (3.00)
2 fold	81.50 (1.49)	66.10 (6.61)	14.80 (4.66)	79.87 (2.77)	77.54 (1.38)	7.70 (3.06)
5 fold	82.25 (1.67)	**70.26 (5.59)**	15.60 (6.42)	79.87 (2.77)	77.27 (1.78)	6.90 (3.93)
10 fold	82.15 (1.63)	69.48 (4.21)	15.50 (6.29)	79.87 (2.77)	77.22 (1.77)	7.20 (2.39)
	Waveform			German		
Classifier	Train (%)	Test (%)	RVs	Train (%)	Test (%)	RVs
fRVM	100 (0)	**89.42 (0.59)**	35.90 (5.13)	98.60 (0.48)	**76.13 (1.92)**	137.90 (4.20)
MOEA	94.83 (1.15)	88.47 (0.73)	28.90 (4.38)	79.44 (0.96)	75.13 (2.23)	30.10 (6.31)
2 Fold	93.27 (1.27)	88.38 (0.75)	18.30 (4.90)	78.63 (0.88)	75.80 (2.36)	17.70 (4.64)
5 Fold	93.68 (0.93)	88.34 (0.91)	18.70 (8.27)	78.77 (0.94)	75.23 (2.35)	24.90 (7.19)
10 fold	93.53 (1.30)	88.36 (0.43)	18.90 (4.93)	78.79 (0.96)	75.37 (2.72	22.00 (7.23)
	Pima			MNIST 5 v All		
Classifier	Train (%)	Test (%)	RVs	Train (%)	Test (%)	RVs
fRVM	93.25 (1.92)	**75.93 (1.32)**	31.40 (4.27)	100	**97.36**	61
MOEA	87.05 (1.82)	75.00 (2.89)	26.20 (7.48)	96.70	95.36	28
2 Fold	83.55 (2.89)	75.27 (2.16)	12.00 (4.47)	97.30	96.13	35
5 Fold	84.10 (2.65)	75.51 (2.45)	14.80 (4.59)	96.40	93.84	17
10 fold	84.65 (2.40)	75.75 (2.24)	14.60 (4.20)	95.70	93.88	15

Table 2: Average training and test accuracies and number of relevance vectors used for the fRVM and EA variants for solutions with highest accuracy in training archive. Numbers in brackets are standard deviations over 10 folds. Underlined test rates are at the significantly different at the 5% significance level using a Mann-Whitney-Wilcoxon test.

	Banana		Breast Cancer		Titanic		Waveform		German		Pima		MNIST: 5 v All	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
fRVM	0.99	**0.95**	0.98	0.68	0.68	0.64	1.00	0.94	1.00	0.75	0.98	0.81	1.00	**0.94**
EA RVM	0.97	0.94	0.86	**0.71**	0.73	0.70	0.98	**0.95**	0.82	**0.77**	0.93	**0.82**	0.87	0.90
2 Fold	0.96	0.93	0.82	0.69	0.72	0.70	0.98	**0.95**	0.79	**0.77**	0.88	0.81	0.84	0.91
5 Fold	0.97	0.94	0.83	0.70	0.72	**0.71**	0.98	0.85	0.80	**0.77**	0.89	**0.82**	0.90	0.90
10 Fold	0.97	0.94	0.82	0.70	0.72	0.70	0.98	**0.95**	0.80	**0.77**	0.89	**0.82**	0.83	0.88

Table 3: Average area under the curve for the training and test fronts

evolutionary optimisation, where it is important not to overfit e.g. [15] and we anticipate that this method will be useful for controlling general evolutionary optimisation. Whereas greedy methods may become stuck at local maxima, the stochastic search inherent in evolutionary methods helps in locating the global optimum and importantly yields an entire Pareto front of solutions displaying the optimal trade-offs between true and false positive rates and complexity from which a solution fitting a particular application may be selected.

References

[1] M. Aizerman, E. Braverman, and L. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.

[2] M. Anastasio, M. Kupinski, and R. Nishikawa. Optimization and FROC analysis of rule-based detection schemes using a multiobjective approach. *IEEE Transactions on Medical Imaging*, 17:1089–1093, 1998.

[3] M. Aupetit. Nearly homogeneous multi-partitioning with a deterministic generator. *Neurocomput.*, 72:1379–1389, March 2009.

[4] U. M. Braga-Neto and E. R. Dougherty. Is cross-validation valid for small-sample microarray classification? *Bioinformatics*, 20(3):374–380, 2004.

[5] P. A. Devijver and J. Kittler. *Pattern Recognition: A Statistical Approach*. Prentice-Hall, London, 1982.

[6] A. Faul and M. Tipping. Analysis of sparse bayesian learning. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems*, volume 14, pages 383–389. MIT Press, 2002.

[7] V. Grunet da Fonseca, C. M. Fonseca, and A. O. Hall. Inferential performance assessment of stochastic optimisers and the attainment function. In E. Zitzler, K. Deb, L. Thiele, C. A. Coello Coello, and D. Corne, editors, *First International Conference on Evolutionary Multi-Criterion Optimization*, pages 213–225. Springer-Verlag. Lecture Notes in Computer Science No. 1993, 2001.

[8] M.A. Kupinski and M. Anastasio. Multiobjective genetic optimizaton of diagonostic classifiers with implications ofr generating receiver operating charactistic curves. *IEEE Transactions on Medical Imaging*, 18(8):675–685, 1999.

[9] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278 –2324, November 1998.

[10] D. J. C. MacKay. The evidence framework applied to classification networks. *Neural Computation*, pages 4–720, 1992.

[11] I. T. Nabney. Efficient training of RBF networks for classification. In *ICANN99*, pages 210–215, London, 1999. IEE.

[12] J. J. K. Ó Ruanaidth and W. J. Fitzgerald. *Numerical Bayesian Methods Applied to Signal Processing*. Springer, 1996.

[13] F. Provost and T. Fawcett. Analysis and visualisation of classifier performance: Comparison under imprecise class and cost distributions. In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pages 43–48, Menlo Park, CA, 1997. AAAI Press.

[14] G. Rätsch, T. Onoda, and K.-R. Müller. Soft margins for AdaBoost. *Machine Learning*, 42(3):287–320, 2001.

[15] W. J. Reckhouse, J. E. Fieldsend, and R. M. Everson. Variable interactions and exploring parameter space in an expensive optimisation problem: Optimising short term conflict alert. In *IEEE Congress on Evolutionary Computation*, pages 1–8. IEEE, 2010.

[16] B. D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, 1996.

[17] A. Schmolck and R. M. Everson. Smooth relevance vector machine: a smoothness prior extension of the RVM. *Machine Learning*, 68(2):107–135, 2007.

[18] LinLin Shen, Li Bai, and Michael Fairhurst. Gabor wavelets and general discriminant analysis for face identification and verification. *Image Vision Comput.*, 25:553–563, May 2007.

[19] M. Tipping. The relevance vector machine. In A. Solla, T. Leen, and K. Muller, editors, *Advances in Neural Information Processing Systems*, volume 12, pages 652–658. MIT Press, 2000.

[20] M. E. Tipping. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[21] M. E. Tipping and A. Faul. Fast marginal likelihood maximisation for sparse Bayesian models. In *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, 2003*, 2003.

[22] X. Yao, Y. Liu, and G. Lin. Evolutionary Programming Made Faster. *IEEE Transactions on Evolutionary Computation*, 3(2):82–102, 1999.

[23] E. Zitzler. *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. PhD thesis, Swiss Federal Institute of Technology Zurich (ETH), 1999. Diss ETH No. 13398.

A Classification of Heuristics

Kent McClymont and Zena Wood

University of Exeter

Abstract

The field of heuristic research and optimisation has developed into a broad and diverse range of approaches, encompassing sub-fields such as meta-heuristics and more recently hyper-heuristics. As research in each of these sub-fields progress, existing early classifications become outdated and less relevant. Providing a new formal classification of heuristics will enable researchers to more strictly define the behaviour of heuristics and enable a better understanding of how the approaches relate to one another. In this paper, existing classifications are examined leading to the introduction of a new classification, which covers combinatorial and continuous domains as well as single-, multi- and many-objective problems. The paper concludes with an analysis of the proposed classification as well as a set of examples to illustrate how it may be used by researchers within the field.

1 Introduction

The field of optimisation has developed into a broad field of research comprising numerous methods and approaches. Although each approach is moderately unique, a few shared tenets connect them all. Most, if not all, methods can be seen as an extension of the 'simple' heuristic. The term heuristic is traditionally defined as a partially informed technique for problem solving, often informed through experience. More specifically, in Computer Science and the field of optimisation, the term refers to the class of algorithms that generate solutions to a problem.

Two key concepts are referenced in this definition: problem and solution. Although this may appear to be a simple definition, it can in fact be extended to include elements such as parameters and objective values. Furthermore, these attributes introduce additional terms such as encoding, dimensionality and a range of other features which are explored more fully later in this paper.

Using encoding alone, it is possible to build a classification that examines the types of problems that exist and the methods that are able to solve them. These two key features – the problem and the solver – are clearly interlinked and as such any formal classification or model should reflect this relationship. A variety of classifications have been proposed in the literature covering problems and heuristic methods, focusing on specific paradigms and problem types. For example, whilst the formal model presented in [1] gives a concise account of the basic mechanics of any Evolutionary Algorithm (EA) it is focused on approaches that adopt the analogous model of evolution. These artificial boundaries created in accordance with the analogies associated with each technique are not necessarily a true reflection of the classes of methods available.

Furthermore, as new layers of optimisation emerge, such as the relatively new field of hyper-heuristics, combined with a growing trend to amalgamate methods and produce hybrid optimisers, the boundaries between traditional classes of heuristics, meta-heuristics and memetic methods are becoming less clear. As literature relating to these higher-level paradigms expands and the range of methods they encompasses are more formally defined it is important to supply a compatible classification that defines both the features and scope of the lower-level heuristics to provide a context in which the higher-level aspects can be placed.

In this paper we explore and build upon existing classifications of heuristics and complementary classifications of multi-objective continuous test problems to develop a formal classification of heuristics and meta-heuristics. To illustrate the classification, it is applied to example heuristics and meta-heuristics from the field of Evolutionary Computation. The classification is intended to be independent of any one approach and as such will not explicitly define or classify heuristics by analogy. Instead, variances between methods like Genetic Algorithms (GAs) [2] and Evolution Strategies (ESs) [3] should be evident using the classification, grouping algorithms by features that separate the two classes. The paper concludes with an evaluation of the final classification including comparisons with existing classifications.

2 Relevant Work

Existing literature has proposed classifications and formalised models that cover many aspects and branches of optimisation. These are reviewed below giving special attention to features that we include or extend in our classification.

2.1 Optimisation Problems

In the field of heuristic classification, classifying by applying heuristics, [4][5] examines a range of frameworks for characterising classification problems and analysing heuristic methods for classification. This approach highlights an interesting problem faced in all applications of heuristics: which heuristic best fits a problem and how do you classify both the problem and heuristic in order to best understand their relationship? In addition to the classifications and formalised models relating to the operations of optimising algorithms, it is important to examine the context in which these heuristics are employed (i.e., the problem) for which there exist few classifications.

[6] presents an early classification of test problems, outlining the types of problem by the desired accuracy or tolerance given to the quality of solutions produced by an ideal heuristic. Problems are categorised into optimisation (finding optimal solutions), semi-optimisation problems (finding near-optimal or good solutions), and satisficing (finding qualified solutions). It is this distinction between optimisation and satisficing that is of interest, distinguishing the difference between problems that are easily solvable but are hard to solve well and problems that are hard to solve at all.

Other notable examples of problem classification presented in the literature are those relating to test problems designed to evaluate the performance of the state of the art optimisation methods. From Van Veldhuizen's classification and in-depth analysis of 20th century optimisation test problems [7] to Deb et al.'s early suite of test problems [8] and more recently the review and scalable test problem toolkit presented in [9].

As a result of an extensive literature review, Huband et al. developed a concise and wide covering classification of test problems [9]. This classification, although specifically developed for multi-objective continuous problems, provides a firm foundation from which to extend into a classification for all domains. [9] categorises problems by recommendations and features, giving guidance to the construction of new test problems. Recommendations relate to the desirable attributes of any toolkit and features relate to key characteristics of the test functions. However, the classification can also be decomposed into three primary sets of features: parameter-space geometry; objective-space geometry; and function mapping.

2.2 Heuristics

As mentioned earlier, a heuristic is generally understood to be a method that produces solutions to a problem. This definition is based on a number of anecdotally assumed elements which are briefly outlined here. Optimisation problems are defined as functions that represent a problem and are used to evaluate the quality of solutions. Solutions are most commonly defined as a pair of vectors that represent the solution parameters and the quality of the solution: the objective values. The two concepts can be expanded further: a parameter vector of values can be considered an encoded representation of the solution which is passed to the problem and evaluated, whilst objective values are usually represented as a vector of real numbers returned by the problem that represent the quality of a parameter vector. In this sense, the solution is both the input and related output for a problem. Finally, the encoding defines how to describe a parameter. This is usually binary, combinatorial, integer or real and provides much of the properties of the optimisation problem. The parameter vector may use any number of encodings, but in practice it is common to use just one.

The definitions outlined above gives a context in which each aspect of an optimisation process can be understood but provides little useful information on how to describe a problem or heuristic. For this, formal models and classifications are required.

In [6], a simple model is presented that can be used to describe any heuristic. This model outlines three primary elements of a heuristic: the encoding; the operators or production rules; and the control strategy. In recent literature these elements are known as encoding, operators and selection strategy. Any heuristic can be constructed from this model, and must include these elements. Indeed, more recent models such as the The unified model for multi-objective evolutionary algorithms (UMMEA) [1] can be seen to be extensions of the Generate – Evaluate - Select framework outlined in [6].

2.3 Meta-heuristics

Despite, or perhaps because of, the range of heuristic methods that have been presented in the field of optimisation there exist few general classifications of heuristic methods other than in [6]. Certainly, a range of models and frameworks have been proposed with each new approach or algorithm being given some structure that later variants follow. However, each of the frameworks are limited to specific forms of heuristic and rarely define the boundaries between heuristics and meta-heuristics.

The term meta-heuristic was first introduced by [10] in which it was used to describe higher-level heuristic methods, although no formal definition was provided. [11] introduces a definition by which a meta-heuristic is described as an iterative generation process that guides subordinate heuristics through intelligent strategies and learning mechanisms. This definition is further extended in [12] to allow the potential for the subordinate heuristics to include other "high level procedures." In contrast [13] describes meta-heuristics as biased random search, allowing for the potential of diverging moves to facilitate exploration of the search space, using this intelligent bias to define meta-heuristic search. A summary of these (and other) definitions is given in [14] where the following key features are highlighted: strategies to guide the search; efficient exploration; range from local search to complex learning processes; usually non-deterministic; mechanisms to avoid local minima; not problem specific; use domain specific knowledge in form of low level heuristics; and memetic features to guide the search. Whilst this provides an interesting summary of the various mechanisms that are usually associated with a meta-heuristic, the classes do not accurately describe the features that distinguish meta-heuristic methods from simple heuristics.

In addition to providing a broad survey of meta-heuristic definitions, [14] proposes a new classification of meta-heuristic methods. This classification is based on five orthogonal features, which can be used individually or in combination to classify meta-heuristics. The first distinction is made between nature-inspired and non-nature inspired methods, splitting methods by their conceptual origins, although the authors regard this approach as not very meaningful and difficult to qualify. The second distinction is made between population-based methods and single point (i.e., trajectory) search methods. The third classification is based on dynamic and static objective function methods. This does not refer to dynamic problems but rather meta-heuristics that dynamically and actively alter the objective function to control the search process. A similar distinction is made between meta-heuristics with one or multiple neighbourhood structures, splitting approaches by their use of multiple representations of the search landscape (i.e., different types of fitness function) to aid the search. Finally a distinction is made between methods that use memory and those that do not. Methods that use memory refer to approaches that utilise stored information, like past populations, from beyond the immediate history (i.e., last generation).

2.4 Evolutionary Algorithms

The most prolific field of meta-heuristics is undoubtedly that of Evolutionary Algorithms (EAs). From the earliest examples such as [3], these methods have been shown to effectively employ the iterative stochastic search paradigm, pairing them with easily understood analogies to increase the accessibility of the methods and the field. Many branches of EAs now exist and are too many to describe here. An extensive survey of approaches and the literature can be found in [15].

Whilst there are many different forms of EA, a paper formalising the basic elements of all EAs was introduced in 2000 by Laumanns et al. [1]. The UMMEA provided a simple and general framework for all multi-objective EAs implementing elitist strategies. Using this framework it is possible to highlight specific aspects that distinguish a heuristic from a meta-heuristic. As mentioned earlier, [6] defines a heuristic as a rule of thumb that creates solutions to a problem. [6] goes on to extend this to describe different types of problems and different forms of heuristics, specifically distinguishing the difference between iterative heuristics that are often applied in games and those that simply generate one solution, such as Best First for tree search. In contrast, the UMMEA describes a range of actions beyond the basic operators of an iterative heuristic that affect the selection strategy of the heuristic and control the search process, similar to [14]. These actions calculate and use meta-data to extend traditional selection strategies which are based upon solution-to-solution comparisons. This extension can be used to distinguish between low-level heuristics and high level meta-heuristics.

3 Classification

Whilst a range of approaches have been taken in the literature to formalise the various aspects of optimisation, it would appear that no single classification exists that combines all the elements of a heuristic. Whilst [6] presents a good foundation from which to build a classification, it provides little insight into the various multi-objective problems that are now prevalent in the literature. Likewise, the model presented in [1] gives a concise and comprehensive formal model to describe EAs, but fails to accurately model the lower level heuristics employed in the generation of solutions. Indeed, even recent classifications such as [19] do not comprehensively describe the range of features a heuristic or hyper heuristic may include but instead presents a classification focused on a few select attributes.

These classifications are useful in understanding specialised areas of optimisation but do not give a broad basis for comparison. As outlined at the start of section 2.1, it is important to be able to quantifiably describe both problems and heuristics in order to best select the tool for the job.

In this section we propose a classification of heuristics. The classification is split into three sections: problems, heuristics, and meta-heuristics. Example heuristics classified using this classification are shown in Table 4.

3.1 Problems

Table 1 outlines heuristics features based on the types of problems they are able to solve. These features are designed to interface with counterpart classifications of optimisation problems, allowing for a comparison of heuristic and problem, and so will share some features in existing classifications such as [6] and [9].

Table 1. Classification of Problems

Feature	Description
[P1] Satisficing	Designed to produce valid solutions.
[P2] Continuous	Can operate on real-valued parameters.
[P3] Combinatorial	Can operate on combinatorial encodings.
[P4] Scalable	Can operate on problems with variable length parameter vectors.
[P5] Similar / Uniform domains	The ranges for each parameter are the same.
[P6] Multi-modal	Multiple optimal regions in parameter space.
[P7] Single-objective	Can solve problems with a single-objective.
[P8] Multi-objective	Can solve problems with 2 or 3 objectives.
[P9] Many-objective	Can solve problems with 4+ objectives.
[P10] Noisy	Designed to compensate for noisy problems.
[P11] Dynamic	Designed to operate on dynamic problems.

Following the classification of problems presented in [6], the heuristics are classified by whether they are designed to optimise problems, generating the best possible solution, or solve satisficing problems and generate only feasible solutions. The two problems require very different types of search; the former focusing on quality through tuning parameters and the latter on feasibility through exploration. This is given in [P1], where the two types of problem are considered mutually exclusive – a heuristic either optimises or satisfices.

Features [P2] and [P3] describe the types of encoding the heuristic is designed to operate on. Continuous parameter encodings [P2] refer to values in the range of real numbers and is reflected in heuristics like the additive single-point Gaussian mutation [15] that adds a randomly drawn real-number to the existing parameter value. In contrast, combinatorial parameter encodings [P3] refer to problems that have a discrete, finite set of possible parameter values that may or may not be ordered. This type of encoding is common in industrial problems such as bin packing and is reflected in heuristics such as Best Fit [23]. It is important to note that [P2] and [P3] are not mutually exclusive and it is possible for heuristics to operate on both encodings. This may be either for mixed encoding problems or through encoding independent operations, such as uniform crossover [15] that can be applied to either problem type.

In addition to classifying problems by individual parameter encodings, it is possible to classify a problem by the features of the parameter vector. These features include variable length or scalable problems [P4] where the parameter vector may change in length during the search process. A common example of such an encoding can be found in Genetic Programming (GP) [24] where the length of the chromosome (parameter vector) is directly related to the size of the GP tree.

The feature [P5] which expresses the similarity of parameter ranges is adopted from [9]. Whilst the uniformity of ranges for each parameter may not seem significant, this feature can have a significant effect on methods like Particle Swarm Optimisation (PSO) [25] where, if scaled, biases can be introduced into the areas of the search space, limiting the travel distance in some dimensions. Feature [P6] is also adopted from [9] and is used to distinguish between heuristics that are designed to locate global optima at one mode or optimal solutions at multiple modes. For example, local search methods like Hill Climbers [6] are designed to find optimal solutions at a single mode as opposed to GAs which search for optimal solutions in multiple regions.

In addition to classifying by the parameterisation of problems, it is also common to classify by the dimensionality of the objective space. Features [P7], [P8] and [P9] relate to the different types of objective spaces the heuristics are designed to solve. Whilst single objective methods [P7] like simple hill climbers can only solve single objective problems, it is possible for multi-objective methods [P8], such as GAs, to solve the lower order single objective problem [P7]. Again, the same hierarchy also applies to approaches like MSOPS-II [26] which, whilst specifically designed for hard many-objective problems [P9], could be applied to multi-objective problems [P8].

The feature relating to noisy problems [P10], such as those discussed in [27] and [28], refer to heuristics that are designed to adapt to the class of problems where multiple evaluations of the same solution results in different objective values. Likewise, the dynamic feature [P11] refers to heuristics that are designed to operate on problems where the search space and function mapping changes over time, potentially in response to evaluations [29].

3.2 Heuristics

In this section we outline heuristic features relating to the operations they perform to solve problems, shown in

Table 2. For some heuristics, such as binary mutation in GAs, the solutions will be generated by perturbing the parameters of an existing solution whilst others will construct new solutions without any prior evaluation of the problem, such as Best Fit for the bin packing problem [23]. Clearly, the operations a heuristic performs has a large impact on the information they require and the types of problem they are applicable to. Best Fit, for example, is a very efficient heuristic and is able to operate efficiently in real-time industrial applications whilst GAs are much slower to run and are not well suited to time sensitive domains. The features outlined in this section aim to classify heuristics by the different types of operations they can perform.

The stochastic feature [H1], like the type of heuristic classified by [14], refers to non-deterministic heuristics like mutation in GAs, or indeed the GA itself. Although [14] describe meta-heuristics as iterative control strategies, our classification allows for low-level heuristics with iterative processes (see feature [H2]) as described in [6]. Iterative heuristics [H2] are those methods that generate and evaluate solutions iteratively, such as a hill climber, in contrast to heuristics such as Best Fit [23] that produces one bin packing solution. These types of non-iterating heuristics are often also constructive heuristics [H3], generating new solutions without any input.

Following the classification of [19] this classification also categorises heuristics by constructive [H3] and perturbation [H4] methods, but does not consider these approaches to be mutually exclusive, allowing for the possibility of a heuristic to construct solutions when no suitable input is available and perturb (using a similar mechanism) input solutions when present. An additional form of generative method is also included in this classification: progenic [H5]. Whilst a perturbation heuristic, like mutation, takes a single solution as input, progenic heuristics are defined as those methods that take a population of solutions (often two) as input, producing new solutions based on a combination of the parameters of the input solutions, such as a GA's crossover operator. Again, it is possible for all three features to be present in one heuristic.

The final two features relate to the space the heuristic operates in to generate new solutions. Decision space [H6] refers to methods, such as binary mutation, that generate solutions by creating new parameters directly. Whereas solution space (commonly called phenotypic space) heuristics [H7] refer to the types of operations on decoded solutions, such as GP tree crossover. Solutions (if perturbating or progenic) must be decoded, operated upon and then the new solutions re-encoded into parameter form.

The features outlined in Tables 1 & 2 represent the core classification features that can be applied to all heuristics.

Table 2. Classification of Heuristics

Feature	Description
[H1] Stochastic	The heuristic is non-deterministic, producing variable outputs.
[H2] Iterative	The heuristic iteratively generates multiple solutions.
[H3] Constructive	Solutions are constructed without existing solutions as input.
[H4] Perturbative	Solutions are created by perturbing individual existing solutions.
[H5] Progenic	Solutions are created by combining parameter values from at least two existing solutions.
[H6] Decision Space	Operates on parameter values.
[H7] Solution Space	Operates on problem representations of the parameter values.

3.3 Meta-heuristic

The following features (outlined in Table 3) are extensions of the heuristic classification and inherits all the features of a heuristic. These features describe the structure of a meta-heuristic, the meta-data it uses, as well as adaptive and learning techniques it employs.

The composite feature [Mh1] refers to methods that incorporate other heuristics in their operations. This is a specific model following the subordinate heuristic structure outlined in [14] and is common in EAs such as GAs. This classification does not preclude the possibility of a low-level heuristic (such as Best Fit) to include meta-features, such as learning and/or adaption, and so allows for non-composite methods to be classed as meta-heuristics.

Memetic methods (the use of memory) are adopted from [14] as feature [Mh2] and refers to the class of meta-heuristics that store and use information over iterations and/or applications, such as the archive in Evolution Strategies (ESs) [3]. This archive feature of ESs and the population feature in GAs can also be used to classify meta-heuristics [Mh3]. The population feature describes memetic or meta-populations, where a solution in the population can exist beyond a single iteration. This does not preclude the possibility of a low-level heuristic using populations; assuming no solution persists across multiple generations.

Features [Mh4] and [Mh5] outline the steps in a meta-heuristic in which the meta-data is used. For example, the Adaptive Genetic Algorithm (AGA) [30] uses meta-data in the generation of solutions [Mh4] through the adaption of meta-heuristic parameters [Mh6]. Likewise, the meta-data used in the selection strategy [Mh5] employed in [31] is also an adaptive method [Mh6]. In this classification a distinction is made between adaption [Mh6], which may be a fixed reaction to occurrence of predetermined triggers, and learning; [Mh7] and [Mh8]. As in [19], the use of learning is also split into two categories with methods like the hyper-heuristic Messy

Genetic Algorithms [32] that utilise training phases prior to optimisation being classed as offline learning techniques [Mh7]. In contrast, the online reinforcement learning used in [33] during the optimisation process is classed as online learning [Mh8].

Table 3. Classification of Hyper-heuristics

Feature	Description
[M_h1] Composite	The meta-heuristic incorporates other heuristics in the generation of solutions.
[M_h2] Memetic	Meta-data and solutions are stored over iterations.
[M_h3] Populations	Multiple solutions are produced and evaluated in sets (populations) which can be stored over iterations.
[M_h4] Generation	Meta-data is employed in the generation of solutions.
[M_h5] Selection	Meta-data is used in the selection of solutions for propagation in future iterations.
[M_h6] Parameter Adaption	The meta-heuristic adapts meta-parameters during the optimisation process.
[M_h7] Offline Learning	The meta-heuristic employs offline learning.
[M_h8] Online Learning	The meta-heuristic employs online learning.

Table 4. Example classified heuristics
(● indicates the feature is present in the heuristic)

Feature	Best Fit	Hill Climber	Genetic Algorithm	Particle Swarm Optimisation
[P1] Satisficing	●			
[P2] Continuous		●	●	●
[P3] Combinatorial	●		●	
[P4] Scalable				
[P5] Uniform domains				●
[P6] Multi-modal			●	●
[P7] Single-objective	●	●	●	●
[P8] Multi-objective			●	●
[P9] Many-objective				
[P10] Noisy				
[P11] Dynamic				
[H1] Stochastic		●	●	●
[H2] Iterative		●	●	●
[H3] Constructive	●			
[H4] Perturbation		●	●	●
[H5] Progenic			●	
[H6] Decision Space	●	●	●	●
[H7] Solution Space				
[M_h1] Composite			●	
[M_h2] Memetic			●	●
[M_h3] Populations			●	●
[M_h4] Generation				●
[M_h5] Selection			●	
[M_h6] Parameter Adaption				
[M_h7] Offline Learning				
[M_h8] Online Learning				

4 Discussion

A classification has been presented in Section 3 that incorporates three individual feature sets. An extensive classification system is possible when these are used in combination. Through examples, this section evaluates the limits of this system as well as comparing it to classifications from the literature. Table 4 shows the classification of two heuristics and two meta-heuristics from the literature. The two heuristics are classified using the problem and heuristic feature set, not having any features in the meta-heuristic set.

The first distinction between this classification and the test problem classification presented in [9] is shown by Best Fit, which is classified as a heuristic for satisficing problems. This is because regardless of the problem or the location of the Pareto optimal set in parameter space, the Best Fit algorithm is only guaranteed to produce valid solutions rather than optimal ones. This type of optimisation problem is more common in combinatorial encodings and so is not covered in [9].

The feature sets also highlight the differences between Best Fit and the Hill Climber by the types of encoding they operate on, the iterative process of the Hill Climber and its perturbation nature compared to the constructive Best Fit. Whilst both these types of generation method are covered by [19], the difference between the Hill Climber and the Genetic Algorithm (GA) would not be so apparent, where the GA uses progenic methods to generate solutions. This feature is important as it suggests the combination and sharing of information between solutions, which is not present in the Hill Climber.

The GA and the PSO algorithm are both classified as meta-heuristics due to the presence of meta-heuristic features. The GA and PSO can be differentiated based on a number of features, primarily the encoding and requirement for uniform domains by the PSO. Whilst a PSO is considered a nature inspired technique and often included in the domain of Evolutionary Computation, the lack of progenic generative methods highlights a key distinction between PSOs and most EAs. Whilst the model presented by [1] can be used to model an EA, it cannot be directly applied to the PSO, which limits the comparative information that would be possible to extract from UMMEA models.

Using just these four examples it is possible to demonstrate the wider breadth of heuristics the presented classification can cover compared with existing approaches. In addition, the proposed classification provides a greater depth classification over classifications such as [6], [14] and [19] and can better describe the differences between methods, specifically with regard to the heuristic classification.

However, whilst the classification presented in this paper provides a broad classification of heuristics,

covering a wide spectrum of approaches and levels of abstraction, the selected features are by no means an exhaustive set but rather a selection of salient features in the context of modern optimisers. As research progresses new features will be discovered and explored and as such this classification will require revision and updates to include new features such as whether or not a heuristic is representation invariant [34].

5 Conclusion

This paper explores and builds upon existing classifications of heuristics and complementary classifications of multi-objective continuous test problems to develop a formal classification of heuristics and meta-heuristics. To illustrate the classification, it is applied to example heuristics and meta-heuristics from the field of Evolutionary Computation. The discussion highlights how the new classification provides a wider scope when compared to existing work and provides a basis from which a more detailed heuristic classification could be built. A wide range of salient features have been included to enable a detailed classification of heuristics but there still exist many addition features, such as whether or not a heuristic is representation invariant [34], that could be included. Future work will look to extend this classification where appropriate to include this wider feature set. This will be best informed following a more detailed survey and classification of literature, which should highlight further areas for development.

References

[1] Laumanns, M., Zitzler, E., Thiele, L. A unified model for multi-objective evolutionary algorithms with elitism. In Proc. of the 2000 Congress on Evolutionary Computation. 1 (2000), 46-53.

[2] Goldberg, D. E. Genetic Algorithms in Search Optimization and Machine Learning. Addison Wesley, 1989.

[3] Rechenberg, I. Cybernetic Solution Path of an Experimental Problem. Royal Aircraft Establishment Library Translation, 1965.

[4] Clancey, W. J. Notes on “Heuristic classification”. Artificial intelligence in perspective. MIT Press Cambridge, 1994.

[5] Clancey, W. J. Heuristic classification. Artificial Intelligence. 27, 3 (1985), 289-350.

[6] Pearl, J. Heuristics: Intelligent Search Strategies for Computer Problem Solving. Addison-Wesley, 1983.

[7] Van Veldhuizen, D. A. Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations. PhD thesis, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, 1999.

[8] Deb, K., Thiele, L., Laumanns, M., and Zitzler, E. Scalable Test Problems for Evolutionary Multi-Objective Optimization. Zurich, Switzerland, Tech. Rep. 112, 2001.

[9] Huband, S., Hingston, P., Barone, L., While, L. A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. IEEE Trans. on Evolutionary Computation. 10, 5 (2006), 477-506.

[10] Glover, F. Future paths for integer programming and links to artificial intelligence. Comput. Oper. Res. 13 (1986), 533-549.

[11] Osman, I. H., and Laporte, G. Metaheuristics: A bibliography. Ann. Oper. Res. 63 (1996), 513-623.

[12] Voß, S., Martello, S., Osman, I. H., and Roucairol, C., Eds. Meta-Heuristics-Advances and Trends in Local Search Paradigms for Optimization. Kluwer Academic Publishers, 1999.

[13] Stützle, T. Local Search Algorithms for Combinatorial Problems - Analysis, Algorithms and New Applications. DISKI Dissertationen zur Künstliken Intelligenz. infix, Sankt Augustin, Germany.

[14] Blum, C., and Roli, A. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Computing Surveys. 35, 3 (2003), 268-308.

[15] Coello Coello, C. A., Lamont, G. B., van Veldhuizen, D. A. Evolutionary Algorithms for Solving Multi-Objective Problems. Spinger, 2007.

[16] Cowling, P., Kendall, G., Soubeiga, E. A Hyperheuristic Approach to Scheduling a Sales Summit. In Practice and Theory of Automated Timetabling III : Third International Conference, PATAT 2000. Lecture Notes in Computer Science. Springer, 2079 (2000), 176-190.

[17] Burke, E. K., Kendall, G., Newall, J., Hart, E., Ross, P., and Schulenburg, S. Hyper-heuristics: An Emerging Direction in Modern Search Technology. In Handbook of Metaheuristics, International Series in Operations Research & Management Science. 57 (2003), ch.16, 457-474.

[18] Burke, E. K., Curtois, T., Hyde, M., Kendall, G., Ochoa, G., Petrovic, S., Vazquez-Rodriguez, J. A. HyFlex: A Flexible Framework for the Design and Analysis of Hyper-heuristics. Multidisciplinary International Scheduling Conference (MISTA 2009). Dublin, Ireland, 2009.

[19] Burke, E. K., Hyde, M., Kendall, G., Ochoa, G., Ozcan, E., and Woodward, J. A Classification of Hyper-heuristics Approaches. Handbook of Metaheuristics, International Series in Operations Research & Management Science. Springer, 2009.

[20] Soubeiga, E. Development and Application of Hyperheuristics to Personnel Scheduling. PhD thesis, University of Nottingham, 2003.

[21] Bai, R. An Investigation of Novel Approaches for Optimising Retail Shelf Space Allocation. PhD thesis, University of Nottingham, 2005.

[22] Chakhlevitch, K., and Cowling, P. I. Hyperheuristics: Recent developments. Adaptive and Multilevel Metaheuristics, volume 136 of Studies in Computational Intelligence. Springer, 2008, 3-29.

[23] Burke, E. K., Hyde, M. R., Kendall, G. Evolving bin packing heuristics with genetic programming. LNCS 4193, Proceedings of the 9th International Conference on Parallel Problem Solving from Nature (PPSN'06). 2006, 860-869.

[24] Koza, J.R. Genetic Programming: on the programming of computers by means of natural selection. MIT Press, 1992.

[25] Kennedy, J., Eberhart, R. Particle Swarm Optimization. In Proceedings of IEEE International Conference on Neural Networks IV. 1995, 1942-1948.

[26] Hughes, E. J. MSOPS-II: A general-purpose Many-Objective optimiser. In IEEE Congress on Evolutionary Computation, 2007 (CEC 2007). 2007, 3944-3951.

[27] Fitzpatrick, J. M., Grefenstette, J. J. Genetic algorithms in noisy environments. Machine Learning. 3 (1988), 101-120.

[28] Rolet, P. and Teytaud, O. Adaptive Noisy Optimization. EvoApplications 2010, Part I, LNCS 6024. Spinger-Verlag, 2010, 592-601.

[29] Bryson Jr. A. E. Dynamic optimization. Addison Wesley Longman, 1999.

[30] Srinivas, M., and Patnaik, L. Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Transactions on System, Man and Cybernetics. 24, 4 (1994), 656-667.

[31] Eiben, A. E., Schut, M. C., De Wilde, A. R. Boosting genetic algorithms with self-adaptive selection. In Proc. of the IEEE Congress on Evolutionary Computation. 2006, 1584-1589.

[32] Ross, P., Marín-Blazquez, J. G., Schulenburg, S., Hart, E. Learning a procedure that can solve hard bin-packing problems: A new ga-based approach to hyper-heuristics. In Proceedings of the Genetic and Evolutionary Computation Conference, GECCO 2003. Springer, 2003, 1295-1306

[33] Mabu, S., Hirasawa, K., and Hu, J. A Graph-Based Evolutionary Algorithm: Genetic Network Programming (GNP) and Its Extension Using Reinforcement Learning. Evolutionary Computation. 15, 3 (2007), 369-398.

[34] Rowe, J. E., Vose, M. D., and Wright, A. H., Representation Invariant Genetic Operators, Evolutionary Computation. 18, 4 (2010), 635-660.