Proceedings of the

Seventh York Doctoral Symposium

on

Computer Science & Electronics

Department of Computer Science & Department of Electronics
The University of York
York, UK

30th October 2014
Editor: Matt Luckcuck

Copyright © 2014 the authors.
Sponsoring Institutions

The Organising Committee are extremely grateful to the following institutions for generously sponsoring YDS 2014. Without such sponsorship YDS would not be able to provide doctoral students of Computer Science and Electronics the opportunity to experience such a well rounded, fully fledged academic conference in their field as a training experience.

Academic Sponsorship

The University of York

The Organising Committee would like to thank both the Departments of Computer Science and Electronics, and the Juniper and DreamCloud research projects for their sponsorship.
Industrial Sponsorship

The Organising Committee would like to thank the following organisations for their sponsorship.

IBM

RAPITA Systems Ltd.

THALES

ETAS
Preface

The York Doctoral Symposium on Computer Science and Electronics (YDS) is an international post-graduate student symposium, now in its seventh year. YDS 2014 was again organised jointly between the departments of Computer Science and Electronics. Here we built upon the foundations laid by last year’s Organising Committee and I hope that this partnership continues next year. The primary aims of the symposium are to provide an opportunity for early-stage researchers to observe and take part in the process of academic publication through organising a full-scale, international academic event, and to allow doctoral students from the UK and Europe to experience a peer-reviewed conference where they can share and exchange their research and ideas in a supportive environment of their peers. Furthermore, by bringing together students from a wide range of areas, YDS also hopes to promote more interdisciplinary research.

YDS 2014 offered three categories of submissions, full-length papers, extended abstracts and posters. We received 3 extended abstracts, 8 full-length papers, and 15 posters – including 2 posters from research projects at the University of York that had sponsored YDS 2014. We saw a slight decrease in the number of submissions when compared to last year. However, 7 of the 12 full-length papers and extended abstracts that we received were from external universities, with 1 of those submissions originating from the Eindhoven University of Technology in the Netherlands. Further, 2 of the 15 posters we received were from external universities. Unfortunately we received a disproportionally low number of submissions from the University of York’s Department of Electronics, with only 1 of the papers and 4 of the posters being submitted by electronics students. Interestingly, 1 submission was received from the University of York’s Environment department via the York Centre for Complex Systems Analysis. The acceptance rate for full-length papers and extended abstracts combined was 67%, including one submission which we accepted but was withdrawn by the author as the date of YDS 2014 clashed with his MSc graduation. It was important that the YDS Programme Committee ensured that all submitted work was reviewed with due anonymity and fairness, as the event is a serious training exercise for the organisers as well as for the contributors and attendees. Therefore, both full-length papers and extended abstract received three anonymous reviews each; poster abstracts received one review each.

We are very grateful to our sponsors without whom YDS could not take place. YDS 2014 was sponsored by the Departments of Electronics and Computer Science at the University of York; our industrial sponsors: IBM, Rapita Systems, ETAS, and Thales; and two research projects in the Department of Computer Science, Juniper and DreamCloud. Among other things, their financial help enabled us to offer prizes for the best full-length paper and extended abstract,
the best presentation, the best poster judged by the industry panel, and the
best poster voted for by attendees. We were honoured to host invited keynote
talks by Dr Ian Broster, from Rapita Systems, Prof. Andy Marvin, from the
University of York, who kindly stepped in at the last minute when one of our
keynote speakers took ill; and Dr Leandro Indrusiak, from the University of York.

I would like to express my gratitude to the YDS 2014 Organising Committee for
their work on the design and logistics of the event, particularly James Stovold,
the Organising Committee Chair, and Alena Denisova for their invaluable and
near tireless work; to the members of the Programme Committee for their time
and concentration in producing such useful and professional reviews, particularly
Pedro Ribeiro, Andrew Turner, and Nils Morozs; to the academic staff at the
University of York for their promotion and support of YDS, especially Dr Mike
Dodds and Prof. John Clark; to the Computer Science administration staff who
were always helpful and cheerful, no matter what ridiculous things we needed
help with or had delivered to the department reception; to Camilla Danese in
the Department of Electronics for all her help disseminating information to her
department’s students; and to the University of York for giving us the opportu-
nity to run YDS. I would like to extend my personal thanks to my supervisors
Prof. Ana Cavalcanti, who compelled me to apply to be the Programme Chair
for YDS 2014, and Prof. Andy Wellings, whose response to my telling him that
I had agreed to be the Programme Chair of YDS 2014 was utterly priceless.

YDS 2014 has been an exhausting, stressful, and personally difficult endeav-
our. However, I believe that the chance to chair the Programme Committee of
an academic conference is a worthwhile experience, one that I am glad that I
did not pass up. YDS has allowed me to gain useful career skills while forging
several personal friendships along the way. I would like to echo the sentiments
of Sam Simpson, the YDS 2013 Programme Chair, in encouraging all post-
graduate students to take part in YDS; whether that be in organising, reviewing
for, contributing to, or simply attending YDS. Finally, I would like to wish the
Organising Committee for YDS 2015 all the luck and determination that they
will undoubtedly require.

Matt Luckcuck
YDS 2014 Programme Chair

‘The road goes ever on and on, down from the door where it began...’

— J.R.R. Tolkein
Organisation

All members of the YDS 2014 Organising and Programme Committees are from the University of York, UK.

Organising Committee

Chairs
James Stovold

Committee
Sophie Alexander
Theo Cho
Alena Denisova
Matthew Luckcuck
Hashan Mendis
Tautvydas Mickus
James Stovold
Bharath Sudev
Christopher Timperley
**Programme Committee**

**Chairs**

Matt Luckcuck  
Theo Cho

**Committee**

<table>
<thead>
<tr>
<th>Stewart Aitken</th>
<th>Sophie Alexander</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antonio Arnone</td>
<td>Chris Bak</td>
</tr>
<tr>
<td>Konstantinos Barmpis</td>
<td>Edward Bowles</td>
</tr>
<tr>
<td>Prof. Alan Burns</td>
<td>Prof. Alister Burr</td>
</tr>
<tr>
<td>Theo Cho</td>
<td>Gabriel Costa Silva</td>
</tr>
<tr>
<td>Alena Denisova</td>
<td>Mike Dodds</td>
</tr>
<tr>
<td>Prof. Edwin Hancock</td>
<td>Prof. David Howard</td>
</tr>
<tr>
<td>Russell Joyce</td>
<td>Sorachai Kamollinsakul</td>
</tr>
<tr>
<td>Ali Karami</td>
<td>Prof. Tim Kelly</td>
</tr>
<tr>
<td>Selahattin Kosunalp</td>
<td>Anna Ladi</td>
</tr>
<tr>
<td>Matt Luckcuck</td>
<td>Prof. John McDermid</td>
</tr>
<tr>
<td>Hashan Roshantha Mendis</td>
<td>Jenny Michel</td>
</tr>
<tr>
<td>Tautvydas Mickus</td>
<td>Nils Morozs</td>
</tr>
<tr>
<td>Prof. Helen Petrie</td>
<td>Joseph Pugh</td>
</tr>
<tr>
<td>Pedro Ribeiro</td>
<td>Prof. John Robinson</td>
</tr>
<tr>
<td>Prof. Colin Runciman</td>
<td>Adolfo Sanchez-Barbudo Herrera</td>
</tr>
<tr>
<td>Sam Simpson</td>
<td>Frank Soboczenski</td>
</tr>
<tr>
<td>James Stovold</td>
<td>Bharath Sudev</td>
</tr>
<tr>
<td>Prof. Jon Timmis</td>
<td>Chris Timperley</td>
</tr>
<tr>
<td>Andrew Turner</td>
<td>Prof. Andy Tyrrell</td>
</tr>
<tr>
<td>Michael Walker</td>
<td>Prof. Jim Woodcock</td>
</tr>
<tr>
<td>Yan Yan</td>
<td>Athanasios Zolotas</td>
</tr>
</tbody>
</table>
## Contents

### I  Keynote Talks

<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Life After Doctorate – Experience Starting and Growing a Technology</td>
<td>3</td>
</tr>
<tr>
<td>Dr. Ian Broster</td>
<td></td>
</tr>
<tr>
<td>Electromagnetic Metrology. New Techniques for Shielding Measurements</td>
<td>4</td>
</tr>
<tr>
<td>Prof. Andy Marvin</td>
<td></td>
</tr>
<tr>
<td>Bio-inspired Resource Management in Multiprocessor and Distributed</td>
<td>5</td>
</tr>
<tr>
<td>Computing</td>
<td></td>
</tr>
<tr>
<td>Dr. Leandro Soares Indrusiak</td>
<td></td>
</tr>
</tbody>
</table>

### II  Full-Length Papers

<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic Time Multiplexed Virtual Channels, a Performance Scalable</td>
<td></td>
</tr>
<tr>
<td>Approach in Network-On-Chip Routers to Reduce Packet Starvation</td>
<td>9</td>
</tr>
<tr>
<td>Bharath Sudev and Leandro Soares Indrusiak</td>
<td></td>
</tr>
<tr>
<td>Improved Traffic Prediction Accuracy in Public Transport Using</td>
<td></td>
</tr>
<tr>
<td>Trusted Information in Social Networks</td>
<td>19</td>
</tr>
<tr>
<td>Ahmad Faisal Abidin, Mario Kolberg, and Amir Hussam</td>
<td></td>
</tr>
<tr>
<td>Labelling Images without Classifiers</td>
<td></td>
</tr>
<tr>
<td>Theodore Boyd and Lewis D. Griffin</td>
<td>27</td>
</tr>
<tr>
<td>Recurrent Cartesian Genetic Programming Applied to Famous Mathematical Sequences</td>
<td>37</td>
</tr>
<tr>
<td>Andrew Turner and Julian Miller</td>
<td></td>
</tr>
<tr>
<td>Evaluation of ADFD and ADFD+ techniques</td>
<td>47</td>
</tr>
<tr>
<td>Mian Asbat Ahmad and Manuel Oriol</td>
<td></td>
</tr>
<tr>
<td>Motrusca: interactive model transformation use case repository</td>
<td></td>
</tr>
<tr>
<td>Joost van Pinxten and Twan Basten</td>
<td>57</td>
</tr>
</tbody>
</table>

### III  Extended Abstracts

<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Too Many Questionnaires: Measuring Player Experience Whilst Playing Digital Games</td>
<td>69</td>
</tr>
<tr>
<td>Aliimran Nordin, Alena Denisova, and Paul Cairns</td>
<td></td>
</tr>
</tbody>
</table>
IV Poster Presentation Abstracts

High Capacity Cognitive LTE Networks for Temporary Events
Nils Morozs, Tim Clarke, David Grace

An Experimental Programme to Measure the Total Power Radiated by an Arcing Pantograph Using a Reverberation Chamber
Lan Mu, Andy Marvin, Yinghong Wen, Eva Karadimou, and Rob Armstrong

Key based fault monitoring in Network-on-Chip interconnection systems for enabling fault tolerant or fault evasive countermeasures
Bharath Sudev

The JUNIPER Project
Neil Audsley, Andy Wellings, Ian Gray, Jamie Garside, and Yu Chan

Identity of Players in Digital Games
Amelia Jati Robert Jupit and Paul Cairns

Efficient Hybrid Impulse Response Synthesis for Room Acoustic Modelling
Stephen Oxnard and Damian Murphy

DreamCloud: Dynamic Resource Allocation in Many-core Embedded and High Performance Systems
Leandro Soares Indrusiak, Neil Audsley, Piotr Dziurzanski, and Amit Kumar Singh

NOMAD Non-Standard Operand Mechanisms Architecture Demonstrator
Christopher Crispin-Bailey and Savan Vachhani

Automatic Paraphrase Extraction for Modern Standard Arabic
Amal Alshahrani

Tools that Encourage Migration to Cloud Computing: A Comparison between Adopters and Non-Adopters
Mohammed Almanea

Diagnosing Parkinson’s Disease with Evolutionary Algorithms
Stuart Lacy
Development of an efficient model to increase public awareness on air pollution

Gabor Makrai, Prof. Peter Cowling, Steve Cinderby, and Stephen Remde

Accelerating the Linux Virtual File System

Russell Joyce

Monitoring Water Pollution with a Heterogeneous Robotic Swarm

Mayank Parmar, Steven Johnson, Jon Timmis

Compact Low Phase Noise Oscillator at 3.8GHz

Pratik D Deshpande and Jeremy Everard
Part I

Keynote Talks
Life After Doctorate Experience Starting and Growing a Technology Spinout

Dr Ian Broster

Rapita systems

Biography Dr Ian Broster is a founder and Director of Rapita Systems, a company who specialise in real-time software analysis. He gained his PhD from the University of York and was based in the Real-Time Systems group during his time in academia. He will be presenting a keynote speech on the work done by Rapita Systems and how academic research feeds into industry.
Electromagnetic Metrology. New Techniques for Shielding Measurements

Prof. Andy Marvin
University of York

Biography  Professor Andy Marvin is the leader of the Physical Layer Research Group in the University of York’s Department of Electronics. In 2011 he became the first serving member of the Department of Electronics to be elected a Fellow of Royal Academy of Engineering. The year before he was elected a Fellow of IEEE. Professor Marvin's research interest is in metrology, mainly in electromagnetic compatibility (EMC). He kindly stepped in to cover for a keynote speaker who had to pull out of YDS due to health reasons and presented a keynote speech entitled Electromagnetic Metrology. New Techniques for Shielding Measurements.
Bio-inspired Resource Management in Multiprocessor and Distributed Computing

Dr Leandro Soares Indrusiak
University of York

Biography Dr Leandro Soares Indrusiak is a Senior Lecturer in the Real-Time Systems group here at the University of York. His research interests include on-chip multiprocessor systems, distributed embedded systems, mapping and scheduling of applications over multiprocessor and distributed platforms, and reconfigurable computing. He presented a keynote speech entitled Bio-inspired Resource Management in Multiprocessor and Distributed Computing.
Part II

Full-Length Papers
Dynamic Time Multiplexed Virtual Channels, a Performance Scalable Approach in Network-On-Chip Routers to Reduce Packet Starvation

Bharath Sudev and Leandro Soares Indrusiak

The Department of Computer Science, The University of York, United Kingdom YO10 5GH
{bs638, leandro.indrusiak}@york.ac.uk

Abstract. Latency performance is enhanced in Network-on-Chips using a variety of techniques, among which virtual channel approach is one of the most widely used. With virtual channels, high priority packets can pre-empt low priority packets but this can cause starvation of low priority packets which can be unnecessary in many situations. With this paper we present an improvement on the classical virtual channel approach which would provide a performance scalability feature that can reduce low priority packet starvation. This new system would allow the router to work in several performance levels so that if high priority packets are received ahead of their deadlines, the performance level of the router can be tweaked so that it would allow lower priority communication interlaced with the high priority communication.

Keywords: Dynamic Time Multiplexed Virtual Channels, Performance Scalability, Packet Starvation Reduction

1 Introduction

With Network-on-Chip (NoC) systems having packets flows with different priority levels, it is important to have hardware to ensure packet predictability based on priority level. With non-preemptive NoC designs, this can be difficult as tailbacking and Head of line blocking of packets can result in undesired magnitudes of latency, even for the highest priority packets. One of the most effective ways to ensure prioritised service is the use of virtual channels where the routers would be able to pre-empt packet flows so that higher priority ones can be transmitted as soon as possible. This is done by introducing multiple service levels for packets so that a higher priority service level packet would be able to pre-empt any lower priority service level communication.

In this paper we present an approach by which such virtual channel based systems can be improved so that starvation of low priority packets can be decreased when possible by interlacing low priority communication with high priority flows.
2 Background

Packet Latency performance can be improved in NoC systems using a variety of techniques and Time Division Multiplexing (TDM) is one of the important ones in this context. TDM is widely used in NoC architectures (like in Aethereal [1] and Nostrum [2]) where each router would have a slot table and the functionality of every router at any point of time would depend on the entry on the respective slot table. TDM provides packet delivery with zero variation in latency but the functionality of the router would be static. For that reason, to enable any new packet flows the whole set of slot tables would have to be re-configured which is complicated and time consuming thus limiting scalability.

Link Division Multiplexing (LDM) introduced by Morgenshtein et al in [3] presents a different approach to improve packet latency predictability. LDM aimed at improving predictability by dividing the communication link into multiple sections so that multiple packets flows would be able to utilise the same link simultaneously. Fig. 1 shows a comparison between TDM based transmission and LDM based transmission and it shows how packet flows A, B and C are treated in both cases.

![Fig. 1. TDM vs LDM](image)

**Fig. 1.** TDM vs LDM (a) TDM based Transmission (b) LDM based Transmission

Fig. 1(a) show the functionality of a TDM based transmission and it can be seen that at each time instant a specific packet is given the full bandwidth of the communication link. For example, if we assume that a flit is transmitted in every clock cycle, on the first clock cycle (T1), packet A is transmitted using the entire bandwidth. At the next clock cycle (T2) B gets transmitted and at the third clock cycle (T3) C gets transmitted.

With LDM however, the whole bandwidth is shared between the three packets A, B and C as shown in Fig. 1(b). Even though multiple packets would be able to utilise
the same link, priority rules can be applied to the transmission by allocating more bus lines (bandwidth) to qualifying packets.

One of the most widely use predictability enhancement techniques is the Virtual Channels [4] approach (like in Nostrum [2] and MANGO [5]). It is an extension to Wormhole flow control aimed at reducing deadlocks, improving network utilisation and for improving performance under congestion. By the use of Virtual Channels, the bandwidth of a physical connection path is multiplexed into separate logical channels so that multiple flits can share the same path.

Introduced by Dally in [19], the virtual channel technique relies on the use of multiple buffers for each channel on the network so that communication through a link is possible even if a flit is blocked on a link.

![Virtual Channel Functionality](image)

**Fig. 2.** Virtual Channel Functionality

**Fig. 2** depicts a comparison between routers without virtual channels and routers with virtual channels. Without virtual channels P2 flits are stuck behind P1 flits when P1 is blocked, as P1 and P2 share ‘node 4’ for transmission. With the use of virtual channels, it can be seen that P2 packets would be able to reach the destination even if the shared node has blocked packets as virtual channels use multiple input buffers for each channel. Mello et al in [6] compared performance of a Hermes [7] NoC with and without virtual channels (by varying the number of virtual channels from 1) and the test reveals reduction of average latency of more than 50% for their 8x8 NoC with uniform load. As virtual channel routers require separate buffers and arbitration/control logic for each service level, virtual channel based routers have high hardware overheads.

There were also predictability enhancement techniques based of adaptive approaches like [8], [9] and [10] where they used dynamically adaptive routing by monitoring traffic in the NoC in real-time. While Ge et al. in [8] utilised a centralised monitoring module to alter the source routing depending on the traffic on the NoC, Cidon et al. in [9] utilised traffic maps in their design for a similar mode of operation. Rantala et al. in [10] dealt with adaptability in a distributed perspective where the source routing at each network interface was altered depending on the congestion information retrieved from neighbouring routers.

With our previous work in [11], [12] and [13], we tried to improve packet predictability by resolving Head of line blocking and tailbacking situations associated which
non-preemptive NoC packets. We used adaptive routers combined with adaptable packets so that Head of line blocking would be resolved by forwarding the priority of the blocked packet to the blocking packet. Tailbacks were resolved by splitting all low priority communication which blocks higher priority communications.

With Dynamic Time Multiplexed Virtual Channels (DTMVC), we are trying to improve upon the virtual channel approach by enabling the router to take account of the packet latency performance so that its performance can be scaled. This would reduce starvation of low priority packets as seen with the classical virtual channel approach. DTMVC can be seen as a dynamic hybrid between TDM and virtual channel approach.

3 Dynamic Time Multiplexed Virtual Channels

With basic virtual channel design, the highest priority active virtual channel gets the entire bandwidth available which can cause starvation of lower priority virtual channel packets. Unless there are late packets of higher priority virtual channels, this kind of extreme approach brings about unnecessary starvation which can be averted. DTMVC provides an opportunity to scale the performance enhancement level of the router so that unnecessary starvation can be avoided. In this paper, we depict the functionality of a router that supports four performance settings.

With DTMVC, the operating time of the design is divided into recurring timeframes. For example, consider a time frame of size 20 clock cycles shown in Fig. 3.

![Fig. 3. Example time frame at PS3](image)

Here we assume that there are four virtual channels; VC0, VC1, VC2 and VC3. The example depicts the lowest of performance level; Performance Setting 3 (PS3) where the time frame is divided into equal time slots for each virtual channel. We can see that in the first quarter of the time frame, Virtual Channel 0 would have the highest priority followed by VC1, VC2 and VC3. During the second quarter, VC1 would get the highest priority followed by VC2, VC3 and VC0. Likewise with the next two
quarters, the rest of the two virtual channels would get equal time slots as the highest priority virtual channel. If the system detects that the latency performance of VC0 is poor resulting in late packets, the router has in built logic to adjust the time frame so that VC0 gets more time slots as the highest priority virtual channel at the cost the slot times allocated to the other channels. So, the router would then switch the Performance Setting to 2 (PS2) thereby modifying the time slot allocation. A possible slot allocation for PS2 is shown in Fig. 4.

![Fig. 4. Example time frame at PS2](image)

Here you can see how VC0 spends more time as the highest priority virtual channel than the previous setting. With subsequent two PS levels, this extension of VC0’s slot time would increase ultimately getting to PS0 where VC0 would always remain as the highest priority channel and the performance would then be like a classical virtual channel design.

![Fig. 5. Example time frame at PS0](image)
The instruction to change performance setting can be provided by a centralised monitor or the packet receptions when late packets are encountered.

3.1 Prototype Architecture

The prototype router was designed as a five port architecture based roughly around Hermes [7] hence employing XY-routing and wormhole switching to reduce hardware requirements. The design uses a uniform mesh topology and unlike Hermes, each packet header includes a priority value which is used by the arbitration unit of the router to resolve contention between packets over output ports. As shown in Fig. 6, the routers have buffered input ports which on reception of a packet header employ XY-routing to set the ‘port request’ register and the ‘priority’ register in accordance with the destination and priority information carried.

![Diagram of input port architecture](image)

Fig. 6. Input port architecture

The arbitration unit in the router then checks ‘port request’ and ‘priority’ registers of all input ports to provide arbitration to the qualified ports. The arbiter then establishes connection by setting the ‘out port’ and ‘flits left’ registers on the input port. This permits the input port to send flits to the allocated output port so that the flits could be transferred away through the communication links. As the flits are being transferred, the input port also decrements the value in the ‘flits left’ register so that when the value reaches zero, the connection can be closed by re-setting the ‘out port’ register value to zero.

For each of the virtual channels supported, this infrastructure is duplicated and there is a ‘control register’ at each input port which designates the active input virtual channel at that moment. To prevent blocking of paths, each input port also has connection lines (for each service level) to the output port of the nearby router to notify the sender of the un-availability of buffer space in the receiver and hence to stop communication. The control logic evaluates the active time frame and other competing flows to enable appropriate input port for data transmission. The priority assignment for the virtual channels is accessed from the slot table so that the appropriate input port can be enabled. The control logic employs a counter to select appropriate
priority assignments at that point of time and an example slot table is provided as Table 1. The router has internal logic to alter the priority assignments inside the slot table so that the performance setting can be altered dynamically.

Table 1. Slot table inside input port at PC3

<table>
<thead>
<tr>
<th>Counter value</th>
<th>Priority of VC0</th>
<th>Priority of VC1</th>
<th>Priority of VC2</th>
<th>Priority of VC3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

3.2 Evaluation

Consider a four virtual channel NoC with four linearly distributed performance settings.

To estimate the effect on the latency with the performance settings PS3, PS2, PS1 and PS0, consider the example in Fig. 7. We assume that there are packets of virtual channel 0, 1, 2 and 3 from the north, west, south and local ports of router R1 destined to router R3. Assuming a packet width of 100 and a packet generation period of zero, the latency performance of the packets can be seen in Fig. 8.

In Fig. 8, it can be seen that at PS0, the whole bandwidth is given to virtual channel zero and hence the latency is seen to be the lowest but this results in starvation of the other service level packets. This is similar to what we see with the classical virtual channel approach. With the performance setting set to one, virtual channel 0 suffers a minor increase in latency as a result of the router allocating time slots to the other service levels. With this setting, packets from virtual channels 1, 2 and 3 gets transmitted but at very high latencies. Similar effect is seen with PS2 and,
with PS3 all the virtual channels are provided equal time slots resulting in equal latency performance.

![Fig. 8. Maximum no contention latency](image)

The elementary prototype was designed in Bluespec System Verilog and in Fig. 9, the hardware comparison with our virtual channel based baseline can be seen.

![Fig. 9. Hardware overhead comparison](image)
The hardware overhead with DTMVC was found to be 40% lookup tables and 14% registers more than the classical virtual channel based NoC. The current prototype has limited functionality as the logic does not account for virtual channels without any transactions, so that a lower priority virtual channel can be provided access even if the slot table depicts the higher priority virtual channel with no flits to transmit. Future work would involve perfecting the HDL model and testing it with standard benchmarks.

4 Conclusion

This paper presented a dynamic technique to decrease packet starvation while using virtual channels in NoC routers. The technique involved splitting the operating time of the router into time slots and then assigning those to each of the virtual channels based on requirement. This allows a scalable performance enhancement system so that starvation of low priority packets happens only when the NoC is unavoidably congested. The paper discussed the technique in detail along with the prototype architecture and did a preliminary investigation of the performance merits and hardware overheads. Future work would involve modifying the prototype to full DTMVC specification and testing it with industry standard benchmarks.

5 References


Improved Traffic Prediction Accuracy in Public Transport Using Trusted Information in Social Networks

Ahmad Faisal Abidin, Mario Kolberg and Amir Hussain

Department of Computing Science and Mathematics, University of Stirling, UK
{faa,mko,ahu}@cs.stir.ac.uk

Abstract. Bus arrival time prediction is a key service for improving public transport attractiveness. In this research, a model of bus arrival time prediction, which improves on the accuracy of such prediction approaches, is proposed. The bus arrival time will be predicted using a Kalman Filter (KF) model which is complemented with additional information from social network communication. Using social network communication in this context, allows to include road traffic information into the prediction, based on people’s experience near the scene. This paper provides an introduction to Kalman Filters and their use for traffic prediction. The paper presents some initial results using a Kalman Filter model for prediction and details how information from social network communication will be added into the Kalman model. Verifying the trustworthiness of social network information is crucial to the success of the approach and the paper discusses some of the challenges.

Keywords: Kalman Filters, Social Networks, Traffic Arrival Prediction, Trust.

1 Introduction

Traffic flow on major urban roads is affected by several factors. It is influenced by a number of conditions and events, such as traffic lights, road conditions, number of vehicles on the road, time of travel, weather conditions, and driving style of vehicles. The provision of timely and accurate travel time information of public transport vehicles is valuable for both drivers and passengers. Recently, arrival time estimation approaches for public transport have attracted an increased interest by researchers applying various paradigms to tackle the problem.

When real-time travel time measurements are available, a dynamic calibration of travel time models is able to improve prediction performance [1]. Dynamic Kalman Filter (KF) models have been shown to not only be able to estimate traffic states on freeways [2], but also to improve the accuracy of urban travel time prediction due to their key property of updating their state continuously creating new observations [1, 3].

There are many algorithms and statistical models that have been proposed for vehicle arrival time prediction. However, there is a gap amongst these algorithms. One particular issue is the question of how the algorithm can receive and incorporate live real-time traffic information. Without receiving such information, algorithms cannot produce an accurate result. If there is an accident 5 miles down the road with a traffic jam which takes 30 minutes to pass, then this information should be included in the arrival time estimation. Very few types of approaches can cope with dynamic information. Kalman Filters is one of them and exhibits the most accurate predictions.

This research proposes an approach which uses information from social networks, especially Twitter, and incorporates this with a Kalman Filter based arrival time prediction algorithm.
Figure 1 depicts the proposed system with its major components. The system takes two types of input: information about the journey (location, speed etc) from the vehicle and road condition updates (accidents, road closures, traffic jams) via social network messages. The system will process the latter to extract the required information from the messages, establish if these messages are trustworthy and if so passes the information together with the journey information into a Kalman Filter model to estimate the arrival time. This is fed back to the vehicle and its passengers and other passenger announcement systems such as bus stop displays.

Using such information with public transport arrival time prediction approaches is novel, and has a major cost advantage over previous approaches employing road sensor data. In addition, this approach allows for the identification of unexpected traffic events, and the subsequent inclusion of this new, real-time information, as part of route calculations and updates. This provides updated information during journeys that may not have been available when travel was initially planned or started. In this situation, social networks can play a pivotal role as an input to the model. Trust is a major concern when using information from social networks. This research proposes to consider trust in the social media information used. We propose to use an approach to trust which takes into account a variety of aspects including location of sender, past behaviour, and reputation in other social networks.

2 Using Kalman Filters for Traffic Arrival Prediction

Recent literature surveys [4][5] have shown that arrival time prediction models are commonly based on historical arrival patterns and/or other explanatory variables correlated with arrival time. The explanatory variables used in these previous studies include historical arrival time (or travel time), schedule adherence, weather conditions, time-of-day, day-of-week, dwell time, number of stops, distance between stops and road–network condition [4][5].

The effect of congestion was treated in various ways in the approaches. For example, some have used traffic properties like volume and speed from simulation results [6], while others have clustered their data into different time periods [7]. Historical data based models were used in geographical areas where traffic congestion is less likely, as the models assumed a cyclical traffic pattern. On the other hand, Kalman Filter techniques and Artificial Neural Network (ANN) approaches were used mostly in urban areas [4]. Importantly, Kalman Filter approaches can be applied with updated input data while the journey is in progress. In fact, besides Kalman Filter approaches, only some statistical approaches [8] take dynamic route information into account.
2.1 Use of Kalman Filters

Kalman Filters are an estimation approach. That is they infer values of parameters from observations which may be noisy, inaccurate, and uncertain. Importantly, unlike many other approaches, Kalman Filters are recursive and hence can take new observations into account as they arrive. With this, Kalman Filters can be executed at runtime of the system under observation. The algorithm is referred to as a ‘filter’ as calculating the estimate from noisy input data essentially ‘filters out’ the noise.

Kalman filters estimate a process by estimating the process state at a given time and then obtaining feedback in the form of noisy measurements. Generally, the equations for the Kalman filter fall into two groups: time update equations and measurement update equations. Time update equations are responsible for predicting forward (in time), using the current state and error covariance estimates to obtain the a priori estimates for the next time step. The measurement update equations are responsible for the feedback, i.e. for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate. The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations. Indeed, the final estimation algorithm resembles that of a predictor-corrector algorithm for solving numerical problems.

A Kalman model implies that the state of a system at a time \( k+1 \) is developed from the previous state at time \( k \). This can be expressed by the following state equation:

\[
x_{k+1} = \alpha x_k + \beta u_k + w_k
\]

(1)

Here \( x_{k+1} \) is the state vector containing the terms of interest (e.g. position, velocity) at time \( k+1 \), \( u_k \) is the vector with the control inputs (e.g. steering angle, braking force) at time \( k \), \( \alpha \) is the state transition matrix which applies the effect of system state parameters at time \( k \) to the state at time \( k+1 \). This basically means that the position and velocity at time \( k \) will affect the position at time \( k+1 \). \( \beta \) is the control input matrix which applies the effects of the control input parameters (\( u_k \)) to the state vector. \( w_k \) is a vector containing process noise terms for each value in the state vector.

Measurements of the system are performed according to the formula:

\[
y_k = \mu x_k + z_k
\]

(2)

where \( y_k \) is the vector of measurements, \( \mu \) is the transformation matrix which maps the state vector parameters to the measurements and \( z_k \) is the vector which contains the measurement noise for each element in the measurement vector. This measurement formula is also referred to as output equation.

If we want to use a standard Kalman filter to estimate a signal, the process that is being measured needs to be described by linear system equations.

2.2 Applying Kalman Filters to Travel Prediction

Kalman filtering models have the potential to adequately accommodate traffic fluctuations with their time-dependent parameters [8-11]. They have been used extensively for predicting bus arrival time [8]. KFs use previous journey data and present observations and measurements as well as a dynamic model, in order to estimate how the 'state' evolves over time and to make future predictions [8,12]. Crucially, Kalman Filters can be applied during an active journey and hence observations of the current journey are included in the estimation. This also allows to include further information...
about the journey to feed into the model. Specifically, in our approach we combine Kalman Filters with information obtained from social networks.

In order to determine the present 'state' of a vehicle, any available data for the position and speed (e.g. from GPS) should be taken into account, however, it should be recognised that none of these observations are perfect. Every measurement has uncertainty, be it due to measurement inaccuracies or sampling issues. These observations would be combined with a 'model' of the travelling behaviour such as the waiting duration if the car stops at traffic lights, or the reduced speed of the car if it slows down due to being near a school. Importantly, the model and the observations are not perfect and the real-world travelling behaviour will not correspond exactly to the model. Thus the model includes dynamic parameters like traffic congestion etc.

Existing arrival time prediction approaches often detect unforeseen hold-ups only with considerable delay. For instance, traffic congestion somewhere on the route is often detected using travel speed information from other vehicles. Consequently, only after a rather large number of vehicles got stuck queuing to get past an accident, is the arrival time estimation updated.

On the other hand, information on road incidents can appear on social networks very rapidly. Thus this paper proposes to integrate such information with Kalman Filter prediction approaches in order to improve the accuracy of the prediction. We propose to compare prediction of conventional GPS based systems with social media information supported systems. As this paper discussed previously, any delay information could initially be 'linearly' added to determine total KF based arrival estimation times. For instance, if a KF model estimated arrival time without external delay information is 10 minutes, and the delay info is 2 minutes, then the updated/total arrival time = 12 minutes.

2.3 Experimentation with a Kalman Filter Model

In order to demonstrate the use of Kalman Filters, this subsection presents a Kalman model and initial experimentation employing it to estimate the position of a car during a journey. In the simulation, it is assumed that the journey takes around 55 minutes, and that the car travels at about 40 km/h for some distance, then slows down to less than 10 km/h and then continues to travel at about 40 km/h until just before it arrives at its destination. Process noise during the journey (delays caused by traffic lights, roadwork etc.) is set at 26 minutes and the noise around 10 minutes. The measurements are updated and fed into the model once every minute.

![Figure 2: Kalman Filter model results.](image-url)
Figure 2 shows the results from the Kalman Filter model. The solid line represents the estimated position by the Kalman Filter whereas the dashed line indicates the measured vehicle position. The vertical lines indicate the velocity of the vehicle. When compared with the measured vehicle location, the curve for the Kalman estimated position is smoothed due to the noise removal. The good accuracy of the model when compared to the measured values is one of the features of the Kalman Filter. In the period between the 13th and 20th minutes, there is a gradual change of velocity from 40 km/h to 10 km/h, possibly due to an accident on the road. The delay is estimated to be around 15 minutes, and then (after 35 minutes) the vehicle continues to move at about 40km/h until the 45th minute when it slows down again before arriving at its destination. In this experiment, the noise/delay is added linearly to system. It is expected that the information pertaining to the delay is received from social network sources (e.g. Twitter).

3 Using Kalman Filters with information from social network

Social networks such as Facebook or Twitter allow users to exchange messages. Especially short messages posted on Twitter reflect events in real time as they happen. Hence, such content is particularly useful for real-time identification of events. Twitter provides hashtags, which allows users to relate their messages to particular events or topics. Specifically, road conditions and incidents are frequently discussed on Twitter. An example of this real-time information is shown in Figure 3.

Figure 3: Information on Social Networks relating to road traffic.

It is important to realise that our approach does not make use of the social network of an individual, but rather gains information from looking at a larger scope, e.g. all messages sent from a certain geographic location such as a city or messages containing particular keywords. Filtering useful information on Twitter in real time is a challenging problem, due to the use of natural language and immense volume of data [13]. In fact extracting all of the information from any Twitter message is an AI-complete problem. Twitter users post messages with a variety of content types, including personal updates and fragments of related information [14]. However, a number of tools exist which analyse and extract information from Twitter messages [15]. Figure 4 shows Topsy as an example. The use of key words or the use of a specific format for Twitter messages including traffic information can help the analysis. Indeed Topsy allows to filter messages according to keywords, hashtags etc. The use of such approaches requires further study.
While travel time data can be obtained through various sources, such as loop detectors, microwave detectors, radar, etc., it is unrealistic to hope that the whole roadway network is completely covered by such data collection devices.

Using Twitter messages such as “M8 Delays of 15min Near J25/J24 westbound” can be leveraged. The information about 15 minutes delay can be input linearly into Kalman Filter models.

Thus besides extracting the actual piece of information from social network messages, a major concern is if the sender and thus the message can be trusted.

4 Using Trust in the System

When receiving messages relating to road traffic in social networks, trusting the sender of messages is crucial. Malicious users may insert bogus messages and distribute false information. Thus a number of criteria can be considered to establish a level of trust in the information. For instance:

- Time of the message (messages sent very soon after the particular event are more valuable, very old messages are out-of-date and counter-productive).
- Location of sender of message (a sender close by the incident is more reliable)
- Sender of the message (messages from a trusted sender are more valuable).

Clearly, the reputation of the sender is important. We propose to use user reputation built up in online social networks as a base to verify the credibility of senders and their messages. Trust within social networks has received a lot of attention recently. An overview can be found in [22]. In our approach, the concepts of Social Trust [16] and degrees of separation [17] are employed to identify the level of trust in messages sent by the sender. To find a trustworthy opinion, relationship and experience are two major features that have to be considered. Thus trust can be determined by the degree of relationship between two people, a closer relationship implies more trustworthy information and opinions. Personal information and experience from previous interactions could be seen as another trust metric [18]. Furthermore, trust can be founded in the roles of users. For example, information from a government agency or the police are seen as trustworthy. In terms of degree of relationship, the number of mutual friends, their behaviour, and relationship history can be used to define a level of trust.

Building trust in social networks, based on popularity and engagement [19], is another way to identify trusted information. Popularity trust refers to the acceptance and approval of a member by others in the community, while engagement trust captures
the involvement of someone in the community. Popularity trust can be seen to reflect the trustworthiness of a member in the community, while engagement trust reflects how much a member trusts others in the community.

The analysis of social media data allows to establish whether particular messages can be used to verify the credibility of messages. Such an analysis can be improved by joining information from different social networks. For instance a Twitter user may have an extensive Facebook or Linked-in network which when combined allows building up a more comprehensive picture. Below we highlight trust measures offered by a number of example social network and e-commerce systems.

**Amazon** uses a ranking system to rank the sellers based on the customer feedback and the number of products sold by the seller on Amazon. Amazon’s seller rating is based on an equation, seller rating = total points/total orders.

**eBay** uses a reputation system to establish trust. eBay collects information on the past behaviour of sellers and buyers. Negative feedback can only be left by buyers. Buyers can leave detailed feedback for sellers in 4 specific categories [20].

**LinkedIn** is a social networking website for people in professional occupations. One purpose of the site is to allow registered users to maintain a list of contact details of people with whom they have some level of relationship. LinkedIn offers a higher degree of linking of people in certain settings. For instance, LinkedIn profiles usually have a larger number of work colleagues as part of a person’s profile than other social media networks. As users often know people in their network personally, this can be used to indicate a higher level of trust.

**Facebook** provides a social network platform for users to connect and to exchange messages. It provides tools to establish the degree of separation between two users. On average, the distance between any two members is 4 degrees [21]. A lower degree of separation indicates a closer acquaintance and thus a higher level of trust.

Finally, there may be some privacy concerns with such a system as users will need to share their current location and information on journeys. Privacy is a complex issue will require further study. However, the system will support appropriate message encryption when sending location and journey information to/from the system. Furthermore, the system will not store such information after the processing is complete. Information shared by users via social networks (road condition information) is regarded to be provided with the knowledge that it can be accessed by other users. Information on group membership and connections/friends on social networks will not be shared with other users.

5 Conclusion

This paper has presented a general framework for a novel approach complementing Kalman Filter models with information from social networks. Kalman Filter models are well established models to predict bus arrival times. The strength of KF models is their ability to predict or estimate the state of a dynamic system from a series of noisy measurements as well as their ability to be executed during a journey. Based on their strength, additional credible and trusted information from social networks can be used to increase the prediction accuracy of KF models. Crucially, as malicious messages can easily be inserted in social networks, only trusted messages from a trusted source must be used. Integrating the information from social networks with the Kalman model, defining a robust approach to trust and ensuring the users’ privacy are key elements of future work.
References

Labelling Images without Classifiers

Theodore Boyd and Lewis D. Griffin
Computer Science, University College London, London WC1E 6BT, UK
{t.boyd,l.griffin}@cs.ucl.ac.uk

Abstract. Verifying that freight containers contain the items listed on their manifests from X-ray scans is a suitable problem for computer vision. However standard techniques do not cope well with the huge numbers of possible categories of cargo, nor with the scarcity of training data for many of these categories. The previously proposed MIRRORING algorithm potentially offers a way to deal with such a problem. The algorithm is based on the Appearance Hypothesis that “words with similar contexts tend to have referents with similar appearance” which allows all training images to be relevant to all labels, by exploiting the full range of this relationship (including dissimilar and intermediate relationships). Previously, this algorithm has only been demonstrated when applied to labelling categories of object, represented by a set of 50 images, with a single label. In this work, we demonstrate the algorithm operating on single images with multiple labels.

Keywords: multi-label, classification, semantics, machine learning, image processing, security science

1 Introduction

At ports of entry (POEs) into a country, governments usually require the inspection of incoming and outgoing goods. At many POEs, freight passing through is imaged by an X-ray scanner like those used for passenger hand luggage but larger and using higher energies, with an X-ray generating linear accelerator on one side and a vertical linear array of detectors on the other [1]. For the high-throughput of a POE, automated analysis of the image would reduce costs and interruptions to the flow of commerce. Such automated analysis could include:

– Detection and localisation of specific objects of interest (e.g. high value items such as vehicles, weapons, narcotics and people) [2].
– Detection of anomalous or unexpected images requiring human inspection.
– Manifest verification: ascertaining whether the contents of the container agree with the legal declaration of its contents in type and amount [3].

This process is currently performed by customs staff at the POE using a combination of targeted image and direct visual inspection. The current paper reports work towards an eventual goal of automating the image-based part of manifest verification.

To develop an automated computer-based manifest verification system we need to overcome two domain-specific problems which prevent usage of standard computer vision recognition methods:
1. There is a large number of categories of object possible: the Harmonised Coding system (an internationally agreed list which defines manifest codes and categories) has $10^6$ possible codes [4].

2. It is not possible to collect adequately sized training sets for each category. Indeed many categories we may encounter have no training examples. This is due to the technology being fairly new, the lack of organised databases and the legal and other difficulties in gaining access to these data.

These two problems lead us to focus on an approach different from the standard one-classifier-per-category, instead we hope to exploit the similarities in appearance between categories. This approach was discussed and implemented successfully for object categories appearing in photographic images by Griffin et al. [5]. We wish to discover whether we can apply these ideas to the current problem. In this present work we build on Griffin et al. [5] by taking two steps towards the full manifest verification problem:

1. Our inputs are single images rather than categories of object (represented by a set of 50 images in Griffin et al.).
2. We use multiple labels per image rather than a single label per category.

To test our algorithms, we use the multi-label classification problem proposed at the 2013 ICML Workshop on Representation Learning [6] and publicly available as a competition on the Kaggle service [7]. The Kaggle challenge training dataset consists of 100,000 photographic images paired with unstructured sets of labels (word bags). The challenge is to learn the relation between images and word bags, so that on a disjoint test set the correct word bag, from a choice of two, can be picked for each image (Fig. 1).

Fig. 1: An example image with the correct and a random incorrect bag of words.

Relating back to freight scanning, the Kaggle images are analogous to freight scans and the word bags to individual manifest files containing multiple codes.

## 2 Object Classification in Computer Vision

Now we briefly discuss possible traditional object classification methods that could be applied to the problem.

In an overview of multi-label classification [8], Tsoumakas and Katakis explain that multi-label classification, previously limited in its scope to text categorisation and medical diagnosis, is now being used in fields as disparate as
protein function classification, music categorisation and semantic scene classification. Our example in Fig. 1 contains a chalk drawing of a boy on a wall. While it would be a fair training example for each of its contained classes individually, it is best described by the multi-label bag \{walk, boy, drawing, \ldots\}, making it suited to the multi-label classification problem.

Another technique that can be used is multiple-instance learning, whereby classification is trained using both positive and negative instances [9]. Negative instances, say Maron and Lozano-Pérez, are those bags of words in which none of the labels are present in the image — therefore, non-matching words are permitted in the positive examples.

There are numerous such general methods for performing multi-label learning and they have been previously summarised thoroughly [10]. We shall consider mainly what is a good baseline method for multi-label classification: training one classifier per label in a “one versus all” fashion, also known as “binary relevance” [11][12]. This method involves producing as outputs predictions for all of the labels for an unseen input whose binary classifiers return a positive match. This is an effective technique and will be the one we shall employ due to its clarity and meaningful foundation. Alternatives include training a classifier for every possible label combination or using dependencies between labels in the form of classifier chains. These multi-label learning methods can be categorised into three major categories [10] [8]. The first is dubbed algorithm adaptation and involves modifying an existing machine learning algorithm for the multi-label problem such as Multi-Label k-Nearest Neighbour (ML-kNN) [13]. The second is problem transformation and works by changing the multi-label problem into a combination of multiple single-label problems which can then be tackled by existing algorithms. Finally, a third idea from Madjarov et al. is that of ensemble methods. They are a hybrid technique that build upon algorithm adaptation and problem transformation methods [10], [8] by either training classifiers on complete subsets of labels (as in RAKEL [14]) or by using additional information such as label dependencies (as in Ensembles of Classifier Chains (ECC) [12]).

3 Distributional Learning of Appearance

Due to the two problems discussed in Section 1, the above described methods are unsuitable for our particular problem. Instead we take a different approach. The major concept used, which has previously been demonstrated [5], is a correlation between word and appearance similarity. Exploitation of this correlation has been proposed as underlying the ability of children to correctly extrapolate labels for more object categories than they have explicitly been taught. Correlation is expressed by the Appearance Hypothesis that “words with similar contexts tend to have referents with similar appearance” [5]. The Appearance Hypothesis derives from a broader, older hypothesis: the Symbol Interdependency Hypothesis that relationships embodied in the world can be found in language (among others, [15]). Using the Appearance Hypothesis, we no longer need to rely on class-specific training sets, instead all images from all classes can contribute to the recognition of each label, which potentially deals with the two problems for manifest verification that we identified.

To make use of the Appearance Hypothesis requires a distance measure between words based on their usage patterns and an image-based distance between...
appearances [16]. The measure of distance between words is termed distributional similarity. An explicit implementation of it is the Correlated Occurrence Analogue to Lexical Semantics (COALS) algorithm [17]. COALS constructs a semantic space of words based on statistical analysis of a corpus of documents in the applicable language. We use the British National Corpus which comprises 97 million words in large collection of written and spoken texts [18]. COALS computes a score for how often a word \( x \) appears in the neighbourhood of a word \( y \), taking account of the frequency of both. These scores are assembled into a distributional vector for each word \( y \). For words \( x \) we use the 14,000 most common (non-stop) words in the corpus. For words \( y \) we use all the labels present in the Kaggle dataset. Distributional vectors between words are compared by correlation, rescaled from \([-1,1]\) to \([1,0]\) so that they are like distances.

For appearance similarity we need image descriptors that are suitable for a set of images which are diverse in context and layout, and that have some traction on semantic context. Histogram methods are suitable for this. There are many to chose from (eg: quantised SIFT [19] and Histogram of Oriented Gradients (HOG) [20]) but following Griffin et al. [5], we use oriented Basic Image Features (oBIFs) and Basic Colour histograms [21] [22].

oBIFs are computed by convolving the input image with a set of six derivative of Gaussian filters [21]. From filter responses, a pixel-by-pixel classification of the image into seven symmetry types is computed (roughly: flat, sloped, minimum, maximum, dark line, light line and saddle point). Quantised orientations are also computed with the calculation depending on the symmetry type. Accounting for symmetry type and orientation yields 23 possible pixel labels. oBIFs are calculated using filters of two different scales, yielding \( 529 = 23^2 \) possible pixel codes. The frequencies of these codes are tallied and normalised into a 529-bin histogram.

Colour histograms are calculated by binning RGB values according to a partition of the colour cube into 11 Basic Colour categories [23], [24]. A histogram of these eleven bins forms the colour feature.

Appearance similarity, or distance, is then defined as \( \arccos(h_i,h_j) \) between two square-rooted normalised histograms \( h_i \) and \( h_j \) in column vector form.

Griffin et al. [5] also describe a MIRRORING algorithm that uses word and appearance distances to assess the compatibility of a label and an image by comparison to a reference set of labelled images. Specifically, the better a label \( w \) is for an image \( I \), the better the correlation between (i) the image distances from \( I \) to the reference set of images, and (ii) the word distances from \( w \) to the labels of the reference set. This is illustrated in Fig. 2.

4 Experiments

Using the approach described in Section 3, the experiments that are reported in this paper are as follows:

1. Baseline MIRRORING implementation. We test whether the algorithm is able to deliver above chance performance even though we apply it to individual images, rather than sets of 50, labelled with a bag of words, rather than a single term. We assess the effect of reference set size, and the performance of oBIF- or colour-based image distance alone or their combination.

2. Comparison of different ways to define build bag-bag distances from word-word distances.
3. Assessment of whether more-frequently occurring words are more useful labels than less-frequently occurring ones.
4. Investigation of whether semantically deep (e.g., “tree”, “leaf”) words are more useful labels than semantically shallow (e.g., “solid”, “organism”) ones.
5. Assessment of whether the effectiveness of a reference set is determined by composition as well as size.

5 Results and Discussion

5.1 Baseline

The MIRRORING algorithm was run against the data provided in the Kaggle dataset. Results are plotted in Fig. 3 for different sized reference sets ($k$), and using oBIFs, colour or their combination. Each point of the plotted curves is performance assessed over $10^4$ trials. Each trial used a different, randomly chosen, test image paired with its correct word bag and a randomly-chosen incorrect word bag. The reference set for each trial was randomly chosen and did not include the test image or the image for the random bag.

Inspection of the results reveals that performance is greater than chance performance (50%) for $k > 2$, rising with $k$ but plateauing around $k = 4096$. Colour performs better than pure oBIFs for all $k$, and the hybrid approach performs the same as colour alone. The maximum performance, achieved at $k = 4096$ is 73.7% using colour histograms. These results contrast with those
in [5] where results plateaued around 660 categories, attaining a score of 77% for colour, 81% for oBIFs and 84% for their combination.

Since the Kaggle competition involved foul play (cheating using the Hungarian algorithm) [6], there is no other baseline against which to compare our work except that of random chance of picking the correct bag (50%).

5.2 Variant Word Bag Distances

The distances between two bags of words should be based on the distances between pairs of words, one from each bag, but there are several ways to do this. A plausible scheme is to use one function \( f_{\text{inner}} \) to compute word-bag distances from word-word distances, and a second function \( f_{\text{outer}} \) to compute a bag-bag distance from the word-bag distances. We have experimented with using minimum, mean, median and maximum to be these two functions.

The results in Table 1 shows the average performance of the classification algorithm using different variants of the \( f_{\text{inner}} \) and \( f_{\text{outer}} \) functions and reference sets of \( k = 32 \) images. The table shows that the best performing methods are the “mean of minimums” or “mean of means”. All other results in this paper (including the baseline results) were computed using the “mean of minimums” method.

5.3 Variant Word Weightings: Frequency

The frequency at which different words appear in a corpus varies over several orders of magnitude [25]. Plausibly, high frequency words may be more tightly bound to the image appearance than low frequency or vice-versa. We assessed this by replacing bags with a single word from the bag either randomly chosen, or the most frequent or least frequent.

For a \( k = 32 \) reference set, the performance using the full bags was 62.1%; using a single random word this dropped, as expected, to 56.1%. The scores using the most frequent (55.7%) and least frequent words (56.2%) were not statistically
Table 1: Mean performance over $10^4$ trials, for $k = 32$ and varying word bag distance methods.

<table>
<thead>
<tr>
<th>$f_{outer}$</th>
<th>Min</th>
<th>Mean</th>
<th>Median</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min</td>
<td>58.7%</td>
<td>60.3%</td>
<td>58.2%</td>
<td>58.8%</td>
</tr>
<tr>
<td>Mean</td>
<td>62.1%</td>
<td>62.1%</td>
<td>60.6%</td>
<td>57.6%</td>
</tr>
<tr>
<td>Median</td>
<td>60.4%</td>
<td>61.7%</td>
<td>60.1%</td>
<td>57.3%</td>
</tr>
<tr>
<td>Max</td>
<td>58.9%</td>
<td>58.1%</td>
<td>57.3%</td>
<td>53.3%</td>
</tr>
</tbody>
</table>

significantly different from the random word, so we conclude that it is not useful to weight word distances according to frequency.

5.4 Variant Word Weightings: Semantic Depth

Semantic depth measures the specificity of words. Low depth words are broad categories (e.g., animal), deeper words are more specific (e.g., squirrel). Like frequency, semantic depth is a plausible feature to use to weight words in distance calculations [26]. We measured semantic depth using path distances from the word ‘entity’ in the WordNet hyponym/hypernym lattice [27]. We first assessed semantic depth using the same random single word method used for frequency.

For a $k = 32$ reference set, and the same baseline full bag performance of 62.1%, using a single random word, this dropped to 55.5%. Then instead using the semantically shallowest word we achieved an increase to 56.7%, while using the deepest word a value of 56.2%. The results shown in Fig. 4(a) support this possible slight improvement for shallow words compared to random ones.

![Fig. 4](image_url)

Fig. 4: (a) Comparing a bag against (i) the semantically shallowest, (ii) semantically deepest, or (iii) a random word in the second bag each time. (b) Varying word weightings by $depth^\alpha$. Baseline performance for the same trials but with no weighting shown by straight line, with 95% confidence intervals in dashed blue.
We further investigated this by using full bag distances, but weighting words according to $depth^\alpha$ for $\alpha \in [-2,2]$. The results shown in Fig. 4(b) show the best effect for $\alpha = -1$, which corresponds to giving greater weight to shallower words.

5.5 Optimal Subsets

We assessed whether all sets of $k$ reference images gave equal performance or whether superior sets could be found. The distribution of performance scores for random $k = 32$ reference sets was found to be approximately normally distributed with $\mu = 62.0\%$ and $\sigma = 1.91\%$, greater than the expected standard variation from $10^4$ trials of $0.49\%$, indicating that there is performance variation which is dependant on subset choice. So we may be able to improve performance by finding effective subsets. We “hunt” for an effective subset starting from a random subset by repeatedly swapping in new random elements and seeing if performance improves. We perform 256 swaps at each different $k$.

![Fig. 5: Performance for random reference sets (blue) compared to effective reference sets discovered by hunting (orange).](image)

Fig. 5 shows that we were able to achieve considerably improved performance by using effective rather than random reference sets. Though the margin between them appears to decrease with increasing $k$.

6 Conclusion

We have demonstrated that the MIRRORING algorithm achieves above chance performance for the Kaggle dataset. Compared to the demonstration of MIRRORING in [5], our experiment used:

2. Multiple labels per image rather than single labels per set of 50 images.
Similarly to [5], we observe increasing performance with the number of reference sets, with an eventual plateau. In contrast to [5], we observed better performance for colour than for oBIF or hybrid. This may be due to the labels having been generated freely in response to the images rather than the images being assembled for a particular label.

In other experiments we determined that “mean of minimums” was the best way to compute bag-bag distances from word-word distances, that frequency weighting was ineffective, that semantically shallow words are slightly more bound to image appearance than deep, and that effective reference sets can be found that are better than random.

Overall, these results show that it is possible to use the Appearance Hypothesis to correctly select a labelling for an image containing multiple classes rather than just one and using only single reference images rather than larger sets.

Relating to the manifest verification problem, this approach shows it is possible to verify whether a bag of words labels a complex scene without having specific classifiers for each and without necessarily having any examples of any individual object. By equating a bag of words to a list of Harmonised Codes within a manifest file, and an X-ray scan to a complex image scene, we have built firm grounding for being able to quantify the labelling match of a manifest file for a cargo scan and thus notice mismatched labelling as is required by the original practical scenario.

References

Recurrent Cartesian Genetic Programming
Applied to Famous Mathematical Sequences

Andrew James Turner¹ and Julian Francis Miller²

¹ The University of York, Electronics Department, andrew.turner@york.ac.uk
² The University of York, Electronics Department, julian.miller@york.ac.uk

Abstract. Recurrent Cartesian Genetic Programming (RCGP) is a recent extension to Cartesian Genetic Programming (CGP) which allows CGP to create recurrent programs. This paper investigates using RCGP to create recurrent symbolic equations which produce famous mathematical sequences. The use of RCGP is contrasted with using standard CGP which can only produce explicit solutions. The results demonstrate that RCGP is capable of producing recursive equations for all the given sequences whereas CGP cannot always produce explicit equations. It is also discussed that since RCGP is a superset of CGP it should also be capable of finding explicit equations. Finally recommendations concerning the initial conditions of RCGP programs are also discussed.

1 Introduction

There are numerous mathematical sequences [12] which have been of interest to mathematicians for many years; as early as Archimedes (287 BC) where mathematical sequences were used in his “Method of Exhaustion” [13]. Mathematical sequences can be defined in two forms, either explicitly or recursively. An explicit equation returns the \( n^{th} \) value in a sequence when passed the value of \( n \) and a recursive equation returns the \( n^{th} \) value in a sequence on the \( n^{th} \) iteration.

Cartesian Genetic Programming (CGP) [6] is a form of Genetic Programming (GP) [4] which has previously being used to map explicit symbolic equations to sequences of numbers [6, 8, 18]. Standard CGP has always been used to produce explicit solutions as it is not capable of creating recurrent equations. However, a recent extension to CGP, Recurrent Cartesian Genetic Programming [16], now enables CGP to create arbitrary programs containing recurrence; including recurrent equations. This paper investigate applying CGP and RCGP to a number of famous mathematical sequences in order to describe them using explicit and recurrent symbolic equations. A separate extension to CGP, Self Modifying CGP, has also previously been used to produce what could described as recurrent solutions [2]. There the CGP chromosomes were able to reconfigure themselves when executed in order to produced different behaviours upon future executions.

The remainder of the paper is structured as follows: Sections 2 and 3 describe CGP and RCGP respectively, Section 4 describes the experiments presented along with the famous mathematical sequences used, Section 5 gives the results of the experiments and finally Sections 6 and 7 give a discussion and present our conclusions.
2 Cartesian Genetic Programming

CGP [6,8] is a form of GP [4] which typically evolves acyclic computational structures of nodes (graphs) indexed by their Cartesian coordinates. CGP does not suffer from bloat\(^3\) [5,15]; a drawback of many GP methods [11]. CGP chromosomes contain non-functioning genes enabling neutral genetic drift during evolution [17]. CGP typically uses point or probabilistic mutation, no crossover and a \((1+\lambda)\)-ES. Although CGP chromosomes are of static size, the number of active nodes varies during evolution enabling variable length phenotypes (solutions). The user therefore specifies a maximum number of nodes, of which only a proportion will be active. Overestimating the number of nodes has shown to greatly aid evolution [7]; which is thought to heighten neutral genetic drift.

Each CGP chromosome is comprised of function genes \((F_i)\), connection genes \((C_i)\) and output genes \((O_i)\). The function genes represent indexes in a function look-up-table and describe the functionality of each node. The connection genes describe the locations from which each node gathers its inputs. For regular acyclic CGP, connection genes may connect a given node to any previous node in the graph, or any of the program inputs. The output genes address any program input or internal node and define which are used as program outputs.

An example of a generic CGP chromosome is given in Equation 1; where \(\alpha\) is the arity of each node, \(n\) is the number of nodes and \(m\) is the number of program outputs. An example CGP program is given with its corresponding chromosome in Figure 1. As can be seen, all nodes are connected to previous nodes or program inputs. Not all program inputs have to be used, enabling evolution to decide which inputs are significant. An advantage of CGP over tree-based GP, again seen in Figure 1, is that node outputs can be reused multiple times, rather than requiring the same value to be recalculated if it is needed again. Finally, not all nodes contribute to the final program output, these represent the inactive nodes which enable neutral genetic drift and make variable length phenotypes possible.

\[
F_0 C_{0,0}...C_{0,\alpha}F_1 C_{1,0}...C_{1,\alpha} .... F_n C_{n,0}...C_{n,\alpha}O_0...O_m
\]  

\(F_0\)\(C_{0,0}\)...\(C_{0,\alpha}\)\(F_1\)\(C_{1,0}\)...\(C_{1,\alpha}\) ... \(F_n\)\(C_{n,0}\)...\(C_{n,\alpha}\)\(O_0\)...\(O_m\)  

(1)

Fig. 1: Example CGP program corresponding to the chromosome: 012 233 124 4

\(^3\) Bloat is a phenomenon seen in many GP methods where the size of the evolved programs grow continuously during evolution with little or no improvement in fitness.
3 Recurrent Cartesian Genetic Programming

In regular CGP chromosome connection genes are restricted to only allow connections to previous nodes in the graph. In RCGP this restriction is lifted so as to allow connections genes to connect a given node to any other node (including itself) or program inputs. Once the acyclic restriction is removed, RCGP solutions can contain recurrent connections or feedback. An example RCGP program is given in Figure 2 along with its corresponding chromosome. Another simpler but less flexible method of using CGP to create recurrent programs is to enforce a Jordan type architecture [9]; where program outputs are fed back as inputs. A slightly more complex multi-chromosome version of CGP has also been adapted to be capable of creating transistor circuits which contain cyclic connections [19].

Fig. 2: Example RCGP program cosponsoring to the chromosome: 212 005 134 5

As described in [16], placing no restriction on connections genes results in mutations creating as many recurrent as feed-forward connections. However, it is likely that most problems do not require fifty percent of the connections to be recurrent. For this reason a new parameter was introduced called recurrent connection probability. This parameter controls the probability that a connection gene mutation will create a recurrent connection.

RCGP chromosomes are interpreted identically to standard CGP. The inputs are applied, each node is updated in order of index, and the outputs are read. The next set of inputs are then applied and the process repeated. One important aspect of RCGP, again described in [16], is that it is now possible for a node’s output value to be read before it has been calculated. Here, as described in [16], all nodes are set to output zero until they have calculated their own output value. This is akin to the initial conditions in recursive equations.

4 Experiments

The experiments presented investigate if CGP and RCGP can be used to create explicit and recurrent equations which describe a given sequence. The parameters used by CGP and RCGP are given in Table 1. Although it is known that that using a high number of nodes aids the evolutionary search [7], here a low number of nodes are used so the evolved equations can be easily inspected i.e. are small.

All experiments are implemented and run using the development version\(^4\) of the cross platform open source CGP-Library [14]. A custom fitness function

\(^4\) The development version allows for recurrent connections whereas the current stable release does not.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evolutionary Strategy</td>
<td>(1+4)-ES</td>
</tr>
<tr>
<td>Max Generations</td>
<td>1,000,000</td>
</tr>
<tr>
<td>Nodes</td>
<td>20</td>
</tr>
<tr>
<td>Node Arity</td>
<td>2</td>
</tr>
<tr>
<td>Mutation Type</td>
<td>probabilistic</td>
</tr>
<tr>
<td>Mutation Rate</td>
<td>5%</td>
</tr>
<tr>
<td>Recurrent Connection Probability</td>
<td>10% (for RCGP)</td>
</tr>
<tr>
<td>Function Set</td>
<td>add, sub, mul, div</td>
</tr>
</tbody>
</table>

Table 1: CGP/RCGP Parameters

is used with the CGP-Library when assessing chromosome fitness. This fitness function awards a score equal to the number of values in the training sequence (100), minus the number of correct values predicted in sequence, minus 0.01 for any further correct values after an incorrect value was predicted. Therefore lower fitness represent a fitter solution.

When using CGP, the input to the chromosomes is $n$ and the expected output is the $n^{th}$ value in the sequence. When using RCGP the input is fixed at the value of one and the chromosome is updated multiple times to produce a sequence of numbers. When the chromosome is updated $n$ times it should produce, in order, the first $n$ values in the sequence. It would also be possible to use RCGP and input the value $n$ instead of the constant one. In this case RCGP could produce explicit as well as recurrent solutions. Here however, RCGP was forced to produce recurrent solutions.

The famous mathematical sequences used by the experiments are now introduced.

### 4.1 Hexagonal Numbers

The Hexagonal number sequence, A000384 from [12], is the number of dots which make up a sequence of hexagons and all the hexagons it contains; see Figure 3. It is defined explicitly by Equation 2 where $n \geq 1$. This produces the following sequence: 1,6,15,28,45,66,91,120,153,190,...

$$y(n) = \frac{2n(2n - 1)}{2}$$ (2)

### 4.2 Lazy Caterers Sequence

The Lazy Caterers Sequence (or more formally the central polygonal numbers), A000124 from [12], is the number of pieces a cake can be divided into with $n$ cuts. The sequence is shown graphically in Figure 3 and described explicitly by Equations 3; where $n \geq 0$. This produces the following sequence: 1,2,4,7,11,16,22,29,37,46,...
\[ y(n) = \frac{n^2 + n + 2}{2} \]  

4.3 Magic Constants

The sequence of Magic Constants, A006003 from [12], are the values each row, column and diagonal of a \( n \times n \) magic square\(^5\) can sum to. The magic squares corresponding to \( n=3, 4 \) and \( 5 \) are given in Figure 3. The sequence is described explicitly by Equations 4; where \( n \geq 1 \). This produces the following sequence: 1, 5, 15, 34, 65, 111, 175, 260, 369, 505, ...

\[ y(n) = \frac{n(n^2 + 1)}{2} \]  

4.4 Fibonacci

The Fibonacci sequence, A000045 from [12], is such that each value is the sum of the previous two value; where the first two values are set as one. The sequence is described explicitly in Equation 5, but is more commonly given recursively such as in Equation 6. This produces the following sequence: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...

\[ y(n) = \frac{(1 + \sqrt{5})^n - (1 - \sqrt{5})^n}{2n\sqrt{5}} \]  

\[ y(n) = \begin{cases} 
1, & \text{if } n \leq 1 \\
y(n - 1) + y(n - 2), & \text{otherwise} 
\end{cases} \]  

5 Results

The results of applying CGP and RCGP towards producing number sequences are now presented. All of the results given are the average of fifty independent runs.

5.1 Hexagonal

CGP found an explicit solution to the Hexagonal sequence in all of the fifty runs using an average of 2,481 evaluations\(^6\). The solutions found used an average of 5.44 active nodes. An example explicit solution found using CGP is given in Figure 4. RCGP found a recursive solution to the Hexagonal sequence in all of the fifty runs using a average of 39,279 evaluations. The solutions found used an average of 10.60 active nodes. An example recurrent solution found using RCGP is given in Figure 4.

\(^5\) A magic square is an \( n \times n \) grid of numbers where the sum of each row, column and diagonal is the same value.

\(^6\) The number of evaluations is the number of solutions (chromosomes) evaluated before a solution is found.
5.2 Lazy Caterer

CGP did not find an explicit solution to the Lazy Caterer sequence in any of the fifty runs. However, an explicit solution, using CGP was found in a longer run and is given in Figure 5\(^7\). RCGP found a recursive solution to the Lazy Caterer sequence in 48 of the 50 run using an average of 7,626 evaluations. The solutions found used an average of 10.53 active nodes. An example recurrent solution found using RCGP is given in Figure 5.

5.3 Magic Constants

CGP found an explicit solution to the Magic Constants sequence in all of the fifty runs using an average of 557,592 evaluations. The solutions found used an average of 8.52 active nodes. An example explicit solution found using CGP is given in Figure 6. RCGP found a recursive solution to the Magic Constants sequence in 43 of the 50 runs using a average of 686,929 evaluations. The solutions found used an average of 12.79 active nodes. An example recurrent solution found using RCGP is given in Figure 6.

5.4 Fibonacci

CGP could not find an explicit solution to the Fibonacci sequence in any of the fifty runs. RCGP found a recursive solution to the Fibonacci sequence in all of

\(^7\) Found by repeatedly running CGP with 10,000,000 generations until a solution was found.
the fifty runs using a average of 27,075 evaluations. The solutions found used an average of 12.00 active nodes. An example recurrent solution found using RCGP is given in Figure 7.

As other GP methods have also been previously applied to the Fibonacci sequence a comparison can be made. However it should be noted that the implementations used are very different between methods e.g. the length of the sequences used, the use of training and testing sets, the percentage of runs which found a solution. Therefore only a very superficial comparison can be made. The results of this comparison are given in Table 2 where RCGP is shown to be very competitive.

6 Discussion

For all of the sequences investigated RCGP managed to find recurrent solutions for the majority of runs. CGP however failed to find explicit solutions to the Lazy Caterers sequence and the Fibonacci sequence. It is unsurprising that CGP could
not find a explicit solution for the Fibonacci sequence as it cannot be easily
defined explicitly. However, it was surprising that CGP failed to find explicit
solutions to the Lazy Caterers sequence which can easily be defined explicitly.
It is likely that the task was harder than anticipated and CGP required more
nodes and more generations to find solutions. It was also shown for the Fibonacci
sequence that RCGP produces very competitive results compared to other GP
techniques.

Interestingly not all sequences have explicit forms, for instance many chaotic
sequences do not. Therefore there will be sequences which can never be described
using standard CGP. It is also likely to be true that certain sequences are more
easily described explicitly or recursively. For instance CGP found explicit solutions
much faster for the Hexagonal sequence, whereas RCGP found recurrent
solutions much more easily for the Lazy Caterers sequence; which can both be
easily defined explicitly or recursively. A benefit of RCGP, not explored in this
paper, is that RCGP can create acyclic and cyclic programs. In the research
presented here RCGP was forced to produce recurrent solutions, by fixing the
input, but it is generally capable of producing feed-forward and recurrent so-

Fig. 6: Example CGP and RCGP Magic Constants solutions.

Fig. 7: Example RCGP Fibonacci solution.
Table 2: GP methods: Fibonacci Sequence

<table>
<thead>
<tr>
<th>Method</th>
<th>Evaluations</th>
</tr>
</thead>
<tbody>
<tr>
<td>RCGP</td>
<td>27,075</td>
</tr>
<tr>
<td>Multi-niche Genetic Programming [10]</td>
<td>200,000</td>
</tr>
<tr>
<td>SMCGP [2]</td>
<td>1,000,000</td>
</tr>
<tr>
<td>Machine Language Programs [3]</td>
<td>1,000,000</td>
</tr>
<tr>
<td>Object Oriented GP [1]</td>
<td>20,000,000</td>
</tr>
</tbody>
</table>

It was noticed that many solutions found contained addition nodes with their outputs fed back as an input; such as node (5) in Figure 7. As all nodes are initialised to output zero before they calculate their own output value, this has the effect of implementing a summation; where the output of node (5) is the running sum of all the previous outputs of node (4). This interesting behaviour also hold for subtraction. However it does not hold for multiplication as the node is also initialised to output zero; if a multiplication node’s output were fed back as an input it would forever output zero. It is therefore possible that simpler recurrent equations could be formed if multiplication nodes could be used to store the product of previous inputs; akin to how addition nodes store the summation. This can be achieved by initialising multiplication nodes to output one until they have calculated their own output value. Then multiplication nodes could be arranged such that they produce the product of previous inputs; the same would also be true for division. It is therefore recommended that future developments of RCGP should initialise addition and subtraction nodes to output zero and multiplication and division nodes to output one. It may also be the case that other node functions benefit from being initialised to specific values and this should be considered when extending the function set.

7 Conclusion

This paper has demonstrated the use of RCGP for producing recurrent symbolic equations which describe famous mathematical sequences. It has been shown that CGP is only capable of producing explicit solutions. It has been shown that RCGP is capable of producing recursive solutions and is also capable of producing explicit solutions. It is therefore concluded, given that not all sequences have explicit forms, that RCGP is a more general solution to producing symbolic equations which describe mathematical sequences.

It was also identified that RCGP often arranged addition nodes so as to implement efficient summation operations. Currently, due to all nodes being...
initialised to output zero, this ability does not extend to multiplication nodes producing product operations. Therefore it is recommended that in future work multiplication nodes should be initialised to output one enabling multiplication nodes to produce product operations.

References

Evaluation of ADFD and ADFD\(^{+}\) techniques

Mian Asbat Ahmad and Manuel Oriol
University of York, Department of Computer Science,
Deramore Lane, YO10 5GH YORK, United Kingdom

Abstract. The ever-increasing reliance on software-intensive system is driving research to discover software faults more efficiently. Despite intensive research, very few approaches have studied and used knowledge about fault domains to improve the testing or the feedback given to developers. The shortcoming was addressed by developing ADFD and ADFD\(^{+}\) strategies presented in our previous publications. In the present study, the two strategies were enhanced by integration of the automatic testing tool Daikon and the precision of identifying failure domains was determined through extensive experimental evaluation of real world Java projects contained in Qualitas Corpus. The analyses of results, cross-checked by manual testing indicated that ADFD and ADFD\(^{+}\) techniques are highly effective in providing assistance but are not an alternative to manual testing with the limited available resources.

Keywords: software testing, automated random testing, manual testing, ADFD, ADFD\(^{+}\), Daikon

1 Introduction

The input-domain of a given System Under Test (SUT) can be divided into two sub-domains. The pass-domain comprises of the values for which the software behaves correctly and the failure-domain comprises of the values for which the software behaves incorrectly. Chan et al. [1] observed that input inducing failures are contiguous and form certain geometrical shapes. They divided these into point, block and strip failure-domains as shown in Figure 1. Adaptive Random Testing achieved up to 50\% better performance than random testing by taking into consideration the presence of failure-domains while selecting the test input [2].

![Fig. 1. Failure domains across input domain [1]](image-url)
We have developed two fully automated techniques ADFD [3] and ADFD+ [4], which effectively find failures and failure domains in a specified range and also provide visualisation of the pass and fail domains. The process is accomplished in two steps. In the first step, an upgraded random testing is used to find the failure. In the second step, exhaustive testing is performed in a limited region around the detected failure in order to identify the domains. The ADFD searches in one-dimension and covers longer range than ADFD+ which is more effective in multi-dimension and covers shorter range. Three separate tools including York Extensible Testing Infrastructure (YETI), Daikon and JFreeChart have been used in combination for developing ADFD and ADFD+ techniques. The YETI [5], Daikon [6] and JFreeChart [7] are used for testing the program, generating invariants and plotting the pass and fail domains respectively.

The rest of the paper is organized as follows: § 2 presents enhancement of the techniques. § 3 shows the difference in working mechanism of the two techniques by a motivating example. § 4 highlights the key research questions. § 5 describes the evaluation process comprising experiments, results and answers to the research questions. § 6 presents threats to validity while § 7 points out the related work. Finally, § 8 presents conclusion of the study.

2 Enhancement of the techniques

Prior to experimental evaluation, new features were incorporated in ADFD and ADFD+ techniques to: increase the code coverage, provide information about the identified failure and generate invariants of the detected failure domains as stated below:

1. The GUI was enabled to launch all the strategies defined in YETI from a single interface. As an example, if ADFD strategy is selected for testing, the system automatically hides the field (range value) associated with ADFD+ and displays two fields of lower and upper bounds. On the other hand if ADFD+ strategy is selected for testing, the system automatically hides the two fields (lower and upper bounds) associated with ADFD technique and displays a single field of range value.

2. Code coverage was increased by extending the techniques to support the testing of methods with byte, short, long, double and float type arguments while it was restricted to int type arguments only in the original techniques.

3. Invariants of the detected failure domains were automatically generated by integrating the tool Daikon in the two techniques. Daikon is an automated invariant detector that detects likely invariants in the program [6]. The generated invariants are displayed in GUI at the end of test execution.

4. The screen capture button was added to the GUI to allow the user to capture multiple screen-shots at different intervals of testing for future reference.
3 Difference in working mechanism of the two techniques

Difference in working mechanism of ADFD and ADFD+ for identification of failure domains is illustrated by testing a simple Java program (given below) with the two techniques. It is evident from the program code that failure is generated when the value of variable \( x = \{4, 5, 6, 7 \text{ or } 8\} \) and the corresponding value of variable \( y = \{2, 3 \text{ or } 4\} \). The total number of 12 failing instances form a block failure domain in the input domain.

```java
/**
 * A program with block failure domain.
 * @author (Mian and Manuel)
 */
public class BlockErrorPlotTwoShort {
    public static void blockErrorPlot (int x, int y) {
        if ((x >= 4) && (x <= 8) && (y == 2)) {
            abort(); /* error */
        }
        if ((x >= 5) && (x <= 8) && (y == 3)) {
            abort(); /* error */
        }
        if ((x >= 6) && (x <= 8) && (y == 4)) {
            abort(); /* error */
        }
    }
}
```

The test output generated by ADFD technique is presented in Figure 2. The labelled graph shows 4 out of 12 failing values in red whereas the passing values are shown in blue. The generated invariants identify all but one failing value \( x = 4 \). This is due to the fact that ADFD scans the values in one-dimension around the failure. The test case shows the type of failure, name of the failing class, name of the failing method, values causing the failure and line number of the code causing failure.

The test output generated by ADFD+ technique is presented in Figure 3. The labelled graph correctly shows all the 12 out of 12 available failing values in red whereas the passing values are shown in blue. The invariants correctly represent the failure domain. The test case shows the type of failure, name of the failing class, name of the failing method, values causing the failure and line number of the code causing failure.

The comparative results derived from execution of the two techniques on the developed program indicate that, ADFD+ is more efficient than ADFD in identification of failures in two-dimensional programs. The ADFD and ADFD+ performs equally well in one-dimensional program, but ADFD covers more range around the first failure than ADFD+ and is comparatively economical because it uses fewer resources than ADFD+.
4 Research questions

The following research questions have been addressed in the study:

1. What is the relevance of ADFD and ADFD+ techniques in identification and presentation of failure domains in production software?
2. What types and frequencies of failure domains exist in production software?
3. What is the nature of identified failure domain and how it affects the automated testing techniques?

5 Evaluation

Experimental evaluation of ADFD and ADFD+ techniques was carried out to determine: the effectiveness of the techniques in identifying and presenting the
failure domains, the types and frequencies of failure domains, the nature of error causing a failure domain and the external validity of the results obtained.

5.1 Experiments

In the present experiments, we tested all 106 packages of Qualitas Corpus containing the total of 4000 classes. Qualitas Corpus was selected because it is a database of Java programs that span across the whole set of Java applications and is specially built for empirical research which takes into account a large number of developmental models and programming styles. All packages included in Qualitas Corpus are open source with an easy access to the source code.

For experimental purpose, the main “.jar” file of each package was extracted to get the “.class” files as appropriate input for YETI. All 4000 classes were individually tested. The classes containing one and two-dimensional methods with arguments (int, long, float, byte, double and short) were selected for experimental analysis. Non-numerical arguments and more than two-dimensional methods were ignored because the two proposed techniques support the testing of one and two dimensional methods with numerical arguments. Each test took 40 seconds on the average to complete the execution. The initial 5 seconds were used by YETI to find the first failure while the remaining 35 seconds were jointly consumed by ADFD/ADFD+ technique, JFreeChart and Daikon to identify, draw graph and generate invariants of the failure domains respectively. The machine took approximately 500 hours to perform the experiments completely. Due to the absence of contracts and assertions in the code under test, undeclared exceptions were taken as failures in accordance with the previous studies [8], [3]. The source code of the programs containing failure domains were also evaluated manually to cross-examine the experimental results.

In accordance with Chan et al. [1], classification of failure domain into various types was based on the number of contiguous failures detected in the input-domain as shown in Table 1. If the contiguous failures detected range from 1 to 5, 6 to 49 or 50 and above the failure domain is classified as point, block or strip type respectively. If more than one type of domain are detected in a program, it is termed as mix type.

<table>
<thead>
<tr>
<th>S. No</th>
<th>Type of failure domain</th>
<th>No of contiguous failures</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Point</td>
<td>01 to 05</td>
</tr>
<tr>
<td>2</td>
<td>Block</td>
<td>06 to 49</td>
</tr>
<tr>
<td>3</td>
<td>Strip</td>
<td>50 &amp; above</td>
</tr>
<tr>
<td>4</td>
<td>Mix</td>
<td>point &amp; block</td>
</tr>
<tr>
<td></td>
<td></td>
<td>point &amp; strip</td>
</tr>
<tr>
<td></td>
<td></td>
<td>point, block &amp; strip</td>
</tr>
</tbody>
</table>

Table 1. Classification of failure domains
5.2 Results

The testing of 106 Java packages including 4000 classes, resulted in 25 packages containing 57 classes to have various types of failure domains. The details pertaining to project, class, method, dimension, line of code (LOC) and type of detected failure domains for each class are given in Table 3. Out of the total of 57 methods indicated in the table, 10 methods are two-dimensional while the remaining 47 methods are one-dimensional. A total number of 17262 lines of code spread across 57 classes in various proportions as shown in the table. The results obtained show that out of 57 classes 2 contain point failure domain, 1 contains block failure domain, 50 contain strip failure domain and 4 contain mix failure domain.

Effectiveness of ADFD and ADFD+ techniques

The experimental results confirmed the effectiveness of the techniques by discovering all three types of failure domains (point, block and strip) across the input domain. The results obtained by applying the two automated techniques were verified: by manual analysis of the source code of all 57 classes; by cross checking the test case, the graph and the generated invariants of each class; by comparing the invariants generated by automatic and manual techniques.

The identification of failure domain by both ADFD and ADFD+ is dependant upon the detection of failure by random+ strategy in YETI. Because only after a failure is identified, its neighbouring values are analysed according to the set range to plot the failure domain.

The generation of graph and invariants and the time of test execution directly depends on the range value, if the range value of a technique is greater, the presentation of failure domain is better and the execution time required is higher. This is due to the testing and handling of greater number of test cases when the range is set to a bigger level. Comparatively, ADFD requires fewer resources than ADFD+ therefore it is less influenced by the range value.

Type and Frequency of Failure domains

As evident from the results given in Table 4 - 7, all the three techniques (ADFD, ADFD+ and Manual) detected the presence of strip, point and block types of failure domains in different frequencies. The results obtained show that out of 57 classes 2 contain point failure domain, 1 contains block failure domain, 50 contain strip failure domain and 4 contain mix failure domain. Mix failure domain includes the combination of two or more types of failure domains including point & block, point & strip and point, block & strip.

The discovery of higher number of strip failure domains may be attributed to the fact that a limited time of 5 seconds were set in YETI testing tool for searching the first failure. The ADFD and ADFD+ strategies set in YETI for testing the classes are based on random+ strategy which gives high priority to boundary values, therefore, the search by YETI was prioritised to the boundary area where there were greater chances of occurrence of failures constituting strip failure domain.
Nature of failure domain

The nature of failure domain identified by two automatic techniques (ADFD and ADFD+) and the manual technique was examined in terms of simplicity and complexity by comparing the invariants generated by automatic techniques with those of the manual technique. The results were split into six categories (2 categories per technique) on the basis of simplicity and complexity of failure domains identified by each of the three techniques. The comparative results show that ADFD, ADFD+ and Manual testing can easily detect 56, 48 and 53 and difficultly detect 1, 9 and 4 failure domains respectively as shown in Table 2. The analysis of generated invariants indicate that the failure domains which are simple in nature are easily detectable by both automated and manual techniques while the failure domains which are complex in nature are difficultly detectable by both automated and manual techniques.

Table 2. Simplicity and complexity of Failure Domains (FD) as found by 03 techniques

<table>
<thead>
<tr>
<th>Type of failure domain</th>
<th>No. of classes</th>
<th>EASY to find PD by ADFD</th>
<th>HARD to find PD by ADFD</th>
<th>EASY to find PD by ADFD+</th>
<th>HARD to find PD by ADFD+</th>
<th>EASY to find PD by MT</th>
<th>HARD to find PD by MT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Point</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Block</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Strip</td>
<td>50</td>
<td>50</td>
<td>45</td>
<td>48</td>
<td>0</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Mix</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Mix</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>57</strong></td>
<td><strong>57</strong></td>
<td><strong>48</strong></td>
<td><strong>53</strong></td>
<td><strong>1</strong></td>
<td><strong>9</strong></td>
<td><strong>4</strong></td>
</tr>
</tbody>
</table>

The simplicity of failure domain is illustrated by taking an example of ADFD, ADFD+ and Manual Analysis in Table 7 for class BitSet. The negativeArray failure is detected due to the input of negative value to the method bitSet.of(i). The invariants generated by ADFD are \{i ≤ -1, i ≥ -18\}, by ADFD+ are \{i ≤ -1, i ≥ -512\} and by Manual Analysis are \{i ≤ -1, i ≥ Integer.MIN\}. These results indicate maximum degree of representation of failure domain by Manual Analysis followed by ADFD and ADFD+ respectively. This is mainly due to the bigger range value in manual analysis followed by ADFD and ADFD+ respectively.

The complexity of failure domain is illustrated by taking an example of ADFD, ADFD+ and Manual Analysis in Table 7 for class ArrayStack. The OutOfMemoryError failure is detected due to the input of value to the method ArrayStack(i). The invariants generated by ADFD are \{i ≥ 698000000, i ≤ 698000300\}, by ADFD+ are \{i ≥ 2147483636, I ≤ MAX_INT\}, by Manual analysis \{i ≥ 698000000\}. All the three strategies indicate the same failure but at different intervals. The ADFD+ is unable to show the starting point of failure
due to its small range value. The ADFD easily discovers the breaking point due to its bigger range value while manual testing requires over 50 attempts to find the breaking point.

6 Threats to validity

All packages in Qualitas Corpus were tested by ADFD, ADFD+ and Manual technique in order to minimize the threats to external validity. The Qualitas Corpus contains packages of different: functionality, size, maturity and modification histories.

YETI using ADFD/ADF+ strategy was executed only for 5 seconds to find the first failure in the given SUT. Since both ADFD and ADFD+ are based on random+ strategy having high preference for boundary values, therefore, most of the failures detected are from the boundaries of the input domain. It is quite possible that increasing the test duration of YETI may lead to the discovery of new failures with different failure domain.

A threat to validity is related to the hardware and software resources. For example, the OutOfMemoryError occurs at the value of 6980000 on the machine used for executing the test. On another machine with different specification the failure revealing value can increase or decrease depending on the hardware and software resources.

It is to point out that all non-numerical and more than two-dimensional methods were not considered in the experiments. The failures caught due to error of non-primitive type were also ignored because of the inability of the techniques to present them graphically. Therefore, the results may reflect less number of failures.

7 Related Work

Shape and location of failure domain within the input domain have been studied in the past. Similar to our findings, White et al. [9] reported that the boundary values have more chances of forming strip failure domain. Finally [10] and Bishop [11] found that failure causing inputs form a continuous region inside the input domain. Chan et al. revealed that failure causing values form point, block and strip failure domains [1].

Random testing is quick in execution and experimentally proven to detect errors in programs of various platforms including Windows [12], Unix [13], Java Libraries [14], Heskell [15] and Mac OS [16]. Its potential to become fully automated makes it one of the best choice for developing automated testing tools [17], [14]. AutoTest [18], Jcrasher [17], Eclat [14], Jartege [19], Randoop [20] and YETI [8], [3], [4] are a few of the most common automated random testing tools used by the research community.

In our previous research publications, we have described the fully automated techniques ADFD [3] and ADFD+ [4] for the discovery of failure domains and
have experimentally evaluated the performance with one and two-dimensional error-seeded numerical programs. The current study is a continuation of the previous work. It is aimed at the enhancement of the two techniques for evaluation of the precision of identifying failure domains by integrating Daikon with ADFD and ADFD$^+$. 

Our current approach of evaluation is inspired from several studies in which random testing has been compared with other testing techniques to find the failure finding ability [22], [23], [24]. The automated techniques have been compared with manual techniques in the previous research studies [25], [26]. This study is of special significance because we compared the effectiveness of the techniques by identifying failure domains rather than individual failures considered in the previous studies.

8 Conclusion

Based on the results, it is concluded that the two automated techniques (ADFD and ADFD$^+$) are more effective in identifying and presenting complex (point and block) failure domains with minimal labour. The manual technique is more effective in identifying simple (long strip) failure domain but is tedious and labour intensive. The precision to identify failure domains can be increased by increasing the range value. The results indicate that the automated techniques can be highly effective in providing assistance to manual testing but are not an alternative to manual testing.

Acknowledgments The authors are thankful to the Department of Computer Science, University of York for academic and financial support. Thanks are also extended to Prof. Richard Paige and Prof. John Clark for their valuable guidance, help and cooperation.

References


* Tables 3 - 7 are available at https://code.google.com/p/yeti-test/
Abstract. Modeling and model transformations tools are maturing and are being used in larger and more complex projects. The advantage of a modeling environment and its transformation tools cannot be easily exploited by non-expert users as many subtle intricacies determine the efficiency of transformation languages and their tools. We introduce transformation use case examples that highlight such language/tooling properties. These simple, non-trivial examples have been extracted from an experiment with transformations of Design Space Exploration models. These examples show some typical modeling patterns and we give some insight how to address the examples. We make a case for initiating an interactive, on-line repository for model transformation use cases. This repository is aimed to be example-centric and should facilitate the interaction between end-users and tooling developers, while providing a means for comparing the applicability, expressivity, and efficiency of transformation tools.

Keywords: model transformations, comparison, design space exploration

1 Introduction

Model transformations play a crucial role in Model-Driven Development (MDD) since the OMG Meta-Object Facility [14] has been introduced in 2005. Amongst others, the Eclipse Modeling Framework has eased the creation of fully-featured editors (with Xtext and Sirius) for Domain-Specific Languages (DSLs). This has greatly increased the popularity of MDD in the past years. With increasing maturity of MDD environments [15] one would also expect a similar uptake in usage of model transformation tools. Several model transformation languages and tools have been introduced since 2006, such as QVTo [6], Epsilon [13], and ATL [9] for Eclipse, but also MetaEdit+ MERL [10] and GME’s GReAT [2]. However, we find that there has been limited success in this area.

From our experiences, it is difficult to make an informed decision on what the best approach is for certain model transformation scenarios. Typically, the developer’s experiences with other programming languages and personal preference

*This work was carried out as part of the Octo+ project with Océ Technologies B.V. under the responsibility of TNO-ESI, Embedded Systems Innovation by TNO.
dictate the choice for a particular tool/language. The high level of abstraction makes it difficult to assess the differences between tools and languages, as only basic documentation and examples are available. This information typically focuses only on the strong aspects of a single tool or language.

We therefore recognize the lack of and need for an up-to-date unified public repository for model transformation challenges. To make an informed decision, we need clear, non-trivial use cases with motivating scenarios and corresponding example implementations expressed in several model transformation languages.

Throughout this paper, we use the Design Space Exploration (DSE) scenario as context. DSE is a critical aspect in the design of Embedded Systems as early design decisions can be supported by efficient simulation and analysis of high-level models. This can shorten the time-to-market and reduce late detection of potential performance problems.

We first introduce the motivation, goals, and core ideas behind the Motrusca interactive repository for model transformation use cases (Sec. 2). We then introduce an example motivating scenario (Sec. 3) from which we highlight some example use cases and motivate why they are hard to express succinctly and efficiently in current model transformation tools (Sec. 4). We discuss related work in Sec. 5 and conclude in Sec. 6.

2 Interactive use case repository

With the plethora of model transformation languages and tools available, it has become hard to even enumerate all (Eclipse-based) transformation languages. It is even harder to determine which one can or should be used. It is often unclear what functionality a transformation tool has (or lacks), what the adoption rate is and what the current status of such tools is. Each transformation language comes with its own set of examples and therefore cannot be directly compared to another. Comparing transformation languages and choosing the right one is currently a non-trivial activity.

2.1 Motivation and goals

With the Motrusca interactive model transformation use case repository, we intend to change the way examples are created. Instead of reasoning from the capabilities of a language, we want to reason from the perspective of a use case. Each use case can then be implemented in different model transformation languages, or even different variations in a single model transformation language.

These use cases are the starting point for a repository of good practices in model transformations and discussions between model/transformation developers and tool developers. Although this information can also be captured on forums, StackOverflow, tutorials and blogs, we feel that a dedicated place to

\[ \text{www.eclipse.org/atl/atlTransformations/} \]
\[ \text{www.eclipse.org/epsilon/examples/} \]
\[ \text{www.stackoverflow.com} \]
model transformations is essential to ensure maximum usefulness and visibility of the information. The Transformation Tool Contest (TTC) [7] is a yearly academic event dedicated to a similar goal. However, Motrusca aims to complement their activities by creating continuous, interactive feedback from model transformation user groups.

In short, the Motrusca repository aims to achieve the following goals:

– identify use cases and best practices for model transformations
– trigger interaction between transformation developers and tool developers
– present alternative solutions to model transformation challenges
– provide a means to compare model transformation tooling/languages (e.g. on their efficiency and applicability)

The use cases described in this paper have been added to the repository, along with some initial implementation examples that show some of the differences between Epsilon Transformation Language (ETL), QVTo and ATL.

2.2 Identifying use cases and best practices

New use cases will first be reviewed before implementations can be submitted. Users should be motivated to supply enough information for a minimal example that can be used for discussion. A proper use case example must have at least:

– a user story motivating the context (a scenario)
– a minimal example (meta)model
– a description of what the use case addresses and why the use case is interesting to discuss

Motrusca will aim at making the processes of providing feedback and information as easy and valuable as possible. Transformation developers can then more easily share their experiences and expertise. It gives tool developers insight in the way users think the tool should be used and also leads to exposing novice users to the best practices. Inspired by test-driven development best practices, fallacies of one model transformation tool should be documented and consequently avoided by other tools. We are essentially building a set of generic integration tests that can be leveraged by transformation language/tool developers.

2.3 Interaction between tool users and developers

By providing a view from the perspective of a use case instead of from a tool/language, novice users can become more comfortable with the best practices faster and might be able to choose the proper tooling without the need for much experimentation. One can compare this model to the popular StackOverflow model, where users can ask questions and the community can provide answers and ask questions to clarify the questions. Motrusca will initially aim at Eclipse-based model transformation languages, but the ideas should translate to other modeling platforms as well. Users will be able to vote for the answer/solution that they deem most appropriate to the use case at hand, similar to questions on StackOverflow.
2.4 Providing alternative solutions

The complexity of the model transformation does not have to originate from the transformation language; for example, the source or target meta-model may be inconvenient or inefficient for transformation purposes. Motrusca users are therefore encouraged to provide out-of-the-box thinking and contribute insight into different realizations (of the meta-model) and show the effect on the model transformation use case.

2.5 Comparison overview

It is important that the solutions to the use cases can be compared to one another. A comparison on several aspects based on our motivating case is shown in Fig. 1. It shows several aspects similar to the quality framework of Kolahdouz-Rahimi et. al. [12] to rate the transformation solutions that we have implemented in Java, IncQuery + Xtend and Epsilon. Motrusca users will be motivated to provide feedback in terms of these kind of quality attributes. By aggregating such information, Motrusca can automatically indicate the popularity and effectiveness of transformation tools.

![Fig. 1: Overall comparison of model transformation approaches in DSE](image)

Certain use cases are primarily related to the performance efficiency of the model transformations, which can be properly quantified. The performance can be measured, by making standardized experiments available through a cloud-based service such as Travis\(^6\) or possibly through SHARE [18]. In addition to performance efficiency, the scalability of the runtime is sometimes of interest to be investigated and model instances that reflect scalability issues can be provided with the use cases.

\(^6\)www.travis-ci.org
3 Motivating scenario

The Octopus tool set [4] is developed to support the DSE process that serves as the motivating case study in this paper. To facilitate automatic DSE in the tool set, a Design Space Exploration Intermediate Representation (DSEIR [3]) has been introduced. This formal representation functions as an intermediate between DSLs and a variety of analysis tools (e.g. CPN Tools [8], UPPAAL [5], SDF3 [16]). The use of an intermediate format facilitates re-use of models and tools, and improves model consistency and code maintainability. The Octopus tool set provides generic tools for coping with design decisions in all kinds of embedded systems. The following subsections describe the motivating industrial use case, the philosophy of DSE with Octopus, and the high-level overview of model transformations in Octopus.

3.1 Copier data paths

The primary motivating use case for the Octopus tool set is the DSE for cost-effective software/hardware configurations for the image processing data path in high-performance, high-volume copier applications (Océ Technologies B.V.7). The DSE supports early design decisions where, for example, the type of algorithms and number of processors are explored.

In this domain, the software as well as the execution platform are modeled. The software is modeled with tasks (e.g. scanning, image processing steps, printing) that communicate data and require computation and storage services.

3.2 Design Space Exploration with Octopus

Fig. 2: Separation of concerns in Octopus (from [4]) (left) and Y-chart; separation of platform, application, mapping and diagnostics [11] (from [3]) (right)

7www.oce.com
The Octopus tool set is designed as a modular system (Fig. 2, left) for DSE of software-intensive embedded systems. It provides analysis and simulation services for DSLs that translate to DSEIR. Model transformations play a crucial role in both the analysis plugins and domain specific modeling front ends. The Octopus tool set already supports several analysis/simulation engines, by transformations from DSL to DSEIR and DSEIR to analysis/simulation models.

The model transformations to the analysis plugins are complex mappings of the DSEIR concepts onto different sets of concepts. The transformation from domain specific modeling front end imposes a different requirement; it should support the modeling engineer by providing high-quality feedback on analysis results in terms of the original model components.

3.3 Transformation flow in Design Space Exploration

Octopus contains several modeling front ends that allow the definition of parameterized DSEIR models. These models can be described in terms of a parameterized mapping, application, and platform as indicated in Fig. 2 (right). An application contains tasks, which require a specific service in order to be executed. The platform consists of resources that can provide a service at some rate. The mapping specifies that the required services are allocated on a resource that provides them. The Octopus tool set uses an experiment definition DSL (DseirText) to indicate the design space by stating the (range of) values for the parameters, the model, and type of analysis or simulation to be executed.

---

**Fig. 3: Octopus example transformation orchestration**

The transformation orchestration for DSE with the Octopus tool set is depicted in Fig. 3. It denotes three distinct transformation steps. Step 1 is specific to the modeling front-end, and steps 2 and 3 are specific to an analysis type. For the UPPAAL case, the experiment indicates that the UPPAAL engine should be used to analyze the latency bounds. The transformation engine needs to:

1. create concrete DSEIR models from domain-specific models; e.g. by substituting parameters by values and mapping DSL concepts to DSEIR concepts,
2. transform concrete DSEIR models to UPPAAL models,
3. serialize the UPPAAL model into UPPAAL model/query text.
The transformation from the input format to a DSEIR model consists primarily of copying the structure and binding values to expressions based on the parameter values that are provided in the experiment. The concrete DSEIR model is then transformed into an abstract UPPAAL model and query definition. Finally, the resulting model and queries are serialized into files that serve as the input for the UPPAAL engine.

4 Model transformation use cases

In this section, we highlight a few use cases that we have extracted from the DSE scenario, in particular from the scenario described in Fig. 3. We discuss briefly what difficulties arose when implementing these examples and how different model transformation languages could handle them. We focus on the following Eclipse-based model transformation tools: QVTo, Epsilon, and ATL. For more details on the use cases, as well as models and transformation implementations, see the Motrusca website www.motrusca.net.

4.1 Combining partial specifications into complete specifications

The separation of concerns in DSE inherently leads to multiple partial specifications that can be combined in several ways; Application, Platform, and Mapping (see Fig. 4) are reusable specification parts. Each combination of partial specifications leads to (at least) one concrete output model, which is compiled from the referenced partial specifications. This leads to the following requirements:

1. an arbitrary (unknown) number of target models may need to be produced
2. a set of source elements (e.g. the partial specifications) are transformed into a (set of) target element(s)
3. trace output model elements back to input elements

The first requirement cannot be specified properly in languages like QVTo, ETL, and ATL; the models to be generated need to be enumerated explicitly. We therefore need a parameterizable workflow that repeatedly calls the transformation to generate a specific output model from the input. Such a workflow can only be effective when a single transformation can be executed with a small processing overhead.

The second requirement, considering that multiple output models can be generated from a multiple sources, is inherently difficult to achieve in single-source rules as used in ETL. Before the introduction of EPL (Epsilon Pattern Language), it was impossible to define rules such that a target element could be traced back to the respective source elements. Transformation traces track source to target elements and are used to compute which element must be referenced. The third requirement is therefore an extension to the second requirement. QVTo allows mapping of additional source elements by passing these as parameters to the mapping function. ATL allows definition of multiple source elements in its transformation rules.
4.2 Similar structures

Recall from Section 3.3 that the Octopus toolset defines experiments in a DSL DseirText, which are then exported to the intermediate format DSEIR. The DseirText and DSEIR metamodels in Eclipse are very similar: DseirText contains experiment definitions and parameters, whereas DSEIR does not. The translation between these model instances therefore consists of duplication of most of the structure, with concrete values for the parameterized DseirText expressions. In pure transformation approaches, this will lead to a lot of boilerplate code, as each concrete class in DseirText needs to be transformed into its direct concrete DSEIR counterpart. A higher-order transformation could describe these copy-actions more succinctly.

This problem has been partly addressed in ATL with a refining mode [17] [12] for such transformations. the resulting refined model may overwrite the source model, or can be saved in a new location. Epsilon Flock [13] is designed to migrate between different versions of a metamodel, but can also be applied to similar metamodels, to achieve a similar effect as the ATL refining mode.

4.3 Expanding parameters to concrete values

In DseirText, an Experiment contains a range of values for the parameters required in the SpecificationParts (see Fig. 4). As each SpecificationPart may be defined before any Experiment refers to it, the range of a parameter (ParameterRange) and the actual usage of a Parameter are decoupled. In Java, a simple mapping of parameter name to parameter value can be propagated and queried by expression tree visitors; this makes looking up parameter values inexpensive.

We have been unable to come up with a clean, concise and efficient way to express such transformations. With the Epsilon languages, we can leverage EPL in the ETL context to achieve a similar effect. However, this unnecessarily convolutes the definition of the transformation, as the underlying execution
mechanisms need to be instructed to cooperate, which is not a trivial task. In QVTo, it is possible to look up the defining ParameterRange for a certain encountered Parameter with the $\text{->any}(\text{self.name} = \text{parameter.name})$ operator. This has, however, linear time worst case behavior, in comparison to constant lookup time in the Java implementation.

### 4.4 Semantic gaps in expression trees

The basic concepts in the UPPAAL language are very different from the basic concepts in the DSEIR language and the expression trees transformation is therefore a complex mapping. Consider for example communication between two tasks; in the DSEIR concepts, task A can write directly into the input buffer of task B. However, in UPPAAL, there is no such communication possible and the communication is forced to use a global buffer.

This kind of semantic gap typically leads to many convoluted rules to capture the mapping logic. Expressions related to the buffers have to be considered in a different way than normal expressions; in procedural programming, a parameterized expression visitor can be used. It is, however, non-trivial to come up with a maintainable and efficient solution in model transformation languages.

### 5 Related work

A framework based on the 2001 ISO/IEC software product quality model has been defined for and applied to model transformation to quantify several aspects of model transformation approaches (language and tool combination) [12]. Our work complements this work.

There have been a few attempts at public metamodel [1] and model transformation\(^8\) [7] repositories for gathering insight into usage and performance of transformation languages and tools. The yearly TTC records and compares expert results for a particular model transformation use case [7], providing insight into the status of state-of-the-art model transformation. The results are reported primarily in terms of functionality and appeal. In contrast to these repositories, Motrusca will enable much tighter interaction between transformation and tool developers. Motrusca also enables direct, interactive comparison of transformation languages.

### 6 Conclusions

In this paper, we have introduced a scenario and corresponding use cases that are hard to implement efficiently with the current state-of-the-art model transformation tools. We have highlighted several aspects that are hard for novices to distinguish without having to dive deep into a language. Taking industrial model transformation challenges as a starting point can boost the adoption and

\(^8\)www.eclipse.org/atl/atlTransformations/
advancement of model transformation tools and languages. We introduce the
goals of Motrusca as an interactive on-line repository and motivate its existence.
The presented use cases are a good starting point for the Motrusca repository
at www.motrusca.net.

References

2. Daniel Balasubramanian, Anantha Narayanan, Christopher van Buskirk, and Ga-
obor Karsai. The graph rewriting and transformation language: Great. Electronic
Communications of the EASST, 1, 2007.
3. T. et al. Basten. Model-driven design-space exploration for embedded systems:
4. T. et al. Basten. Model-driven design-space exploration for software-intensive em-
6. Object Management Group. Meta object facility (MOF) 2.0 query/view/transfor-
7. E. Jakumeit, S. Buchwald, D. Wagelaar, L. Dan, Á Hegedüs, M. Hermannsdörfer,
T. Horn, E. Kalhina, C. Krause, K. Lano, M. Lepper, A. Rensink, L.M. Rose,
S. Wätzoldt, and S. Mazanek. A survey and comparison of transformation tools
10. Steven Kelly and Juha-Pekka Tolvanen. Domain-specific modeling: enabling full
11. B. Kienhuis, E. Deprettere, K. Vissers, and P. van der Wolf. An approach for
quantitative analysis of application-specific dataflow architectures. In Proc. ASAP,
February 2014.
15. R.F. Paige and D. Varró. Lessons learned from building model-driven development
18. P. van Gorp and S. Mazanek. Share: a web portal for creating and sharing exe-
cutable research papers. Procedia Computer Science, pages 589 – 597, 2011. ICCS.
Part III

Extended Abstracts
Too Many Questionnaires: Measuring Player Experience Whilst Playing Digital Games

A. Imran Nordin, Alena Denisova, and Paul Cairns

Department of Computer Science, University of York, UK

Abstract. Player experience is an important area of research in digital games. The findings are crucial for the developers, designers and reviewers of games, allowing for the better understanding of player experience whilst playing digital games. Questionnaires are a way to directly measure the reported experiences of players. This approach in games research, however, is challenging for new researchers because of the proliferation of questionnaires available. The problem is knowing which questionnaires are measuring what aspect of experience. This paper sets out the need for positioning the various questionnaires in relation to each other. We list all the current available questionnaires to measure engagement whilst playing digital games. We, therefore, argue that further investigation on these questionnaires is needed to produce better quality questionnaires and reduce confusion amongst player experience researchers.

1 Introduction and Motivation

Terms such as fun, flow, presence and immersion amongst others have been used to describe the experience of playing digital games [1,2]. Though there are many objective (e.g. physiological assessment, such as heart rate measurements, electromyography (EMG), electrodermal activity (EDA), etc.) and subjective (e.g. interviews and focus groups) ways of assessing player experience, they must all at some point reference the subjective nature of individuated experiences. Questionnaires are a useful research method to directly quantify the subjective player experience because they are both easy to deploy and provide a standardised test for quantifying a particular aspect of experience under consideration [3]. Additionally, questionnaires allow players to express their subjective experience albeit within the parameters set by the items of the questionnaire.

Questionnaire is a technique that allows participants to convey their thoughts and feelings within the framework of set questions. These questions act as prompts to the participant – allowing them to consider specific aspects of their gaming experience. More importantly, the written questionnaire also ensures that the same specific aspects are considered by all participants, hence, offering consistency and uniformity.

However, there are a few drawbacks of using questionnaires to measure player experience. Aside from the more obvious problems, such as participants not taking a questionnaire seriously, there is also a less evident and more profound
problem – namely the wording of the questions themselves that reduce the face validity [3], and equally the way in which (and the scale upon which [4]) the participants answer them.

As there are many aspects of player experience, there are many questionnaires that are currently used in order to measure them. Some questionnaires take a broad brush approach looking at all aspects of gaming experience [5,6], others take a specific aspect, such as immersion [7] or motivation [8]. In one sense this proliferation of questionnaires helps researchers to home in on the aspect of concern, but at the same time it can be confusing as to whether questionnaires that purport to measure the same thing (or even apparently different things) are in fact measuring the same thing.

This is not to say that there should only be one questionnaire. The variety in questionnaires is necessary to allow a nuanced focus on different aspects of games. But where, for example, different questionnaires claim to be measuring engagement, they ought to produce consistent and correlated results.

2 Player Engagement Questionnaires

Existing models of player experience use their own questionnaires to measure the overall engagement when playing a game based on certain aspects that influence the game enjoyment. A summary of the most widely used questionnaires is presented in Table 1.

Such a large number of existing questionnaires poses a challenge for new researchers, who may not necessarily be familiar with every specific detail of each theory. Choosing one of them is therefore often based on their availability – many of these questionnaires are not available publicly, or it may be needlessly challenging to obtain some of them. So eventually only those ones that are easily accessible tend to be used for measuring player experience.

Moreover, in order to obtain reliable results, the data needs to be gathered using a reliable questionnaire. Unfortunately, some of existing questionnaires are not statistically validated, and are eventually avoided as they are presumed to not be trustworthy.

Gaming Engagement Questionnaire (GEQ) [5] and Immersive Experience Questionnaire (IEQ) [7], are prominent examples of questionnaires set up in a similar way in order to evaluate player experience. Amongst other questionnaires reviewed in table 1, these two are available publicly and have more similarities between them, than the rest of them. The GEQ was initially developed to assess the impact of deep engagement in violent video games. The questionnaire itself consists of 19 positively worded questions answered on a five-point Likert scale: the higher the score that the user gives for each question, the more engaged they are deemed to be. The formulation of the questionnaire puts engagement on a single dimension that ranges up from immersion to flow. This questionnaire, however, has relatively little empirical validation that has been undertaken to establish the reliability of the questionnaire in part because of its (relatively) recent introduction to the field.
<table>
<thead>
<tr>
<th>Questionnaire Components</th>
<th>Components</th>
</tr>
</thead>
</table>
| **Flow Questionnaire [9]** | Clear goals  
High concentration  
Reduced self-consciousness  
Distorted sense of time  
Direct and immediate feedback  
Balance between ability level and challenge  
A sense of personal control  
Intrinsically rewarding activity |
| **Presence Questionnaire [10]** | Control factor  
Sensory factor  
Distraction  
Realism factor |
| **Immersive Experience Questionnaire (IEQ) [7]** | Emotional involvement  
Cognitive involvement  
Real world dissociation  
Challenge  
Control |
| **GameFlow Questionnaire [11]** | Concentration  
A sense of challenge  
Player skills  
Control  
Clear goals  
Feedback  
Social interaction  
Immersion |
| **Game Engagement Questionnaire (GEQ) [5]** | Absorption  
Flow  
Presence  
Immersion |
| **Player Experience of Needs Satisfaction (PENS) [12]** | Competence  
Autonomy  
Relatedness  
Presence (Immersion) |
| **Social Presence in Gaming Questionnaire (SPGQ) [13]** | Psychological involvement (empathy)  
Psychological involvement (negative feelings)  
Behavioural engagement |

*Table 1. Questionnaires measuring user engagement whilst playing digital games.*
On the other hand, the IEQ is a widely used questionnaire in determining the levels of immersion experienced by players. It has been tested much more empirically across a far-reaching array of different scenarios and game types, for example [14,15,16]. Similarly to the GEQ, it uses five-point Likert scale questions for measuring player experience, but is specifically focused on the notion of immersion when playing games. Unlike the GEQ, the IEQ uses both positive statements and negative statements. Each positively worded statement has a negatively worded counterpart, adding an additional layer of accuracy. The overall score is composed of the summary of the results from the positive questions, as well as the inverted results of the negative ones. The development of the IEQ also suggested that there are five factors underlying immersion, but in practice immersion is also treated as a single dimension with the factors lending aspect for interpretation of results.

3 Proposed Design for the Future Work

As mentioned earlier, several challenges arise, particularly for the new player researchers, when investigating player experience while playing digital games. Usability also appears to be an issue that affects questionnaires [17].

In this position paper, we, therefore, are proposing our idea to investigate two of the main questionnaires used for measuring the experience of playing digital games, namely the Immersive Experience Questionnaire (IEQ) [7] and the Game Engagement Questionnaire (GEQ) [5] to test whether they are both measuring the same experience. Given the similar emphasis of these two questionnaires, it seems reasonable that they are in fact addressing the same underlying aspect of player experience. However, the question is whether this is in fact the case.

Work is underway to test the hypothesis that the IEQ scores are correlated to the GEQ scores. The design of the experiment will involve the manipulation of player experience by using music, as in work of Sanders et al. [16]. Player experience will be measured using both questionnaires, and the results obtained using each questionnaire will be compared. We believe the results will provide an insight to address any potential problem of having multiple measurement tools.
References


Future Media: The Role of HCI in Broadcast

Timothy Neate¹, Matt Jones¹ and Michael Evans²

¹ Department of Computer Science, Swansea University
² User Experience Group, BBC Research and Development

tdjneate@gmail.com

Abstract. Media is changing. The internet has transformed not only the way we communicate, but also the way we ingest media such as TV and radio. This extended abstract describes how technology is currently fuelling change in TV broadcasting and outlines how technologists and content providers will need to adapt to this. We focus on multiscreen experiences, discussing the work in the area and describing its progress and its shortcomings. We then propose a design space for second screen content from a HCI perspective, and discuss some of our progress.

1 Introduction

Multi-screen TV experiences are becoming increasingly prevalent whether the broadcasting corporations instigate it or not. We often browse the web for related content on our secondary devices when watching TV, for example, Googling an actor we recognise in a show when we cannot place him. Moreover, the content we browse is not necessarily related to the content we are viewing on the TV. As discussed by Rooksby et al. [13], the relationship between the two devices becomes a web of related, semi-related, and non-related content; often driven by external factors, such as the social fabric of domestic life.

Such interactions have led to media companies developing second screen content that ties in with the show itself. This has been termed companion content, and can range from the mundane, such as adverts [5], to providing content that attempts to portray an entirely new experience in which the companion content and the show bind together to create a totally new media entity [3]. Regardless of the application, it is clear that the content, both televusal and on the second screen, should bind together into a seamless and complementary experience.

2 Technology and Media

Twenty years ago, when asked to describe TV, most people would report it as an experience in which they receive a show through their TV set in their house. This show, broadcast at a specific time, would consist of an audiovisual experience that would be the same for each viewer. Now, the viewing experience is quite different. The idea of waiting for our favourite show to come on can
often feel a little archaic. As discussed by Vinayagamoorthy et al. [14], the use of IP (Internet Protocol) to send media to our devices has allowed us to view and engage with TV, whenever, wherever, and however we want.

2.1 Object Based Broadcasting

In traditional broadcasting, a TV show is made in the studio – the video, audio, and additional media assets are gathered, and then mixed together as a finalised show and broadcast. Object-based broadcasting, however, does not do this. Each piece of media is gathered as before, but is instead sent individually via IP (Figure 1). This leads to some interesting possibilities for the end user as it can, according to some metadata, be reassembled according to the user’s requirements, as described in more depth on the BBC blog [4].

**Fig. 1:** Object-based broadcasting paradigm, adapted from [4].

Traditional media does not often fit our requirements ideally. When we view media it can be on an array of platforms and scenarios – from mobile devices with tinny headphones on the bus, to home cinemas with 7.1 surround sound. By providing content individually, with associated metadata, it is possible to adapt it to our systems. Not only can content adapt to our devices, but it can also adapt to us and our needs. Moreover, it is possible to use such technologies to embellish the TV experience, for example adapting the audio to a football match to the side of our choice [9], or by providing responsive [6] renditions of stories – intermixing a story with details about the listener, by getting information from their social media feeds, or by analysing the user’s browser data [2, 16].
2.2 Second Screen Experiences

It is evident from work such as [7] that there is a growing trend in using a second screen as a means of interacting with the TV – systems such as the Universal Control API [1] allow for the user to interact with the TV over the home network. This offers a fully configurable remote for interactive content, yielding new possibilities for interaction and empowering the physically disabled with more adaptable user interfaces. As well as catering for the needs of the differently abled, it is also possible to cater for those who surround them by use of object-based broadcasting and dual-screen experiences. A father, who is hard of hearing, need not affect the viewing experience of the family. Additional content such as subtitles, or signing, can be provided on a secondary device without interfering with the family’s viewing experience. Also, the BBC have worked on providing ‘play along’ games for shows such as the Antique Roadshow [15]. In this experience the user is asked to guess the price of a specific object on a secondary device as they watch the show, connecting the two using audio watermarking [10]. This app gained positive feedback, with older viewers in particular embracing the interactive experience.

The pairing of a secondary device and a TV allows for companion content to be provided on the secondary device – content that supplements or embellishes the content on the TV can be provided at specific, or user dictated, moments in the show to enhance the overall televisual viewing experience. In 2010 the BBC undertook some trials of providing supplementary content to a broadcast of the nature show “Autumnwatch”. Additional content, such as facts about the animals in the show, was provided throughout the programme on the users’ secondary devices in real time [8]. In addition to this trial of 300 people, extensive work was done into investigating gaze across two screens in such an experience [3]. Significantly, they found that users pay considerably more attention to the motion-rich TV than the secondary device.

2.3 Design Space

The literature suggests a gap in knowledge in the area of designing user experiences for such scenarios. Though extensive work has been done on potentially interruptive content in task-based environments (a thorough literature review of which is given in [11]), little work has investigated this in a media-based context. With the exception of [12], which studies the implications of integrating an instant messaging service into a TV, the academic literature seems to fall short, especially with regards to second screen experiences. Many questions remain unanswered in this area, and to optimise the broadcasting experience we believe that the following areas should be focused on:

**How the user is alerted to content** – How we alert users to the new content is important for mediating attention between the devices. The alert can be subdivided into modalities, such as auditory, visual, or haptic. Moreover, it can be presented on either device. As discussed earlier, Brown et al. [3] found
that the users focused strongly on the primary screen, which may lead them to miss companion content through change blindness. Ultimately, we believe that the most important thing to consider here is that there is a tradeoff between how well alerted to content the user is, and how immersed they remain in the experience as a whole.

**Context of use** – If we know the context in which the user is viewing the media we can provide content in an appropriate manner. For example, we may provide different types of media in different ways for a user on the train on their phone, compared to someone at home on their sofa, watching TV with companion content on a secondary device.

**Timing** – The time in which content is introduced is important. As discussed in [8] the users preferred new content to arrive on the secondary device in moments of low activity on the primary device. This should be further explored.

**Type of media** – If provided incorrectly the actual content itself will be an interruption. The amount, and the granularity, of content provided in a secondary channel should depend on the timing and the context.

**Interaction** – The level in which the user is expected to interact with the system is important, and the design of the technologies and the content should reflect this. We need to design systems that are not only intuitive, but allow the user to interact while remaining engaged in a TV show.

In summary, there is still a significant amount of research to be done, and technologies developed in the HCI side of this area. It is clear from the literature that this use case is a relatively untouched scenario. We need to explore each of the aforementioned areas and design new methods for providing a way to blend content seamlessly into the currently displayed materials to create a coherent experience.

### 3 Current Progress

We are currently investigating how we best design technology to make the companion content experience more involving for users. To this end we are currently furthering the work of Brown et al. [3] by investigating methods for mediating the attention of users between devices in a typical second screen use case. Our first study shows some strong results that inform the design of such scenarios for technologists and broadcasters alike. In addition to this we have also completed a study into gestural interaction for task-driven scenarios, with an aim to compare the findings to interaction in a typical media-based use case. Future work will involve working on designing new technologies to improve companion content experiences from a technological and broadcasting perspective.
References

Part IV

Poster Presentation
Abstracts
High Capacity Cognitive LTE Networks for Temporary Events

Nils Morozs, Tim Clarke, David Grace
{nm553, tim.clarke, david.grace}@york.ac.uk

Department of Electronics, University of York, York YO10 5DD, UK

Abstract. One of the fundamental tasks of a cellular system is spectrum management, concerned with dividing the available spectrum into a set of resource blocks or subchannels and assigning them to voice calls and data transmissions in a way which would provide a good quality of service (QoS) to the users. Flexible dynamic spectrum access (DSA) techniques play a key role in utilising the given spectrum efficiently. This has given rise to novel wireless communication systems such as cognitive radio networks and cognitive cellular systems. Such networks employ intelligent opportunistic DSA techniques that allow them to dynamically access spectrum underutilized by the incumbent licensed users.

An emerging state-of-the-art technique for intelligent DSA is reinforcement learning (RL); a machine learning technique aimed at building up solutions to decision problems only through trial-and-error. The most widely used RL algorithm in both artificial intelligence and wireless communications domains is Q-learning. Our previously proposed Q-learning based DSA algorithm has been shown to work effectively in scenarios where a cognitive cellular system has to prioritize among resources in its own dedicated spectrum. However, it has not been applied to problems where the spectrum is shared with incumbent systems.

One of the scenarios currently considered in the EU FP7 ABSOLUTE project is a temporary cognitive cellular infrastructure that is deployed in and around a stadium to provide extra capacity and coverage to the mobile subscribers and event organizers involved in a temporary event, e.g. a football match or a concert. The aim of this poster is to demonstrate, using system level simulations, how a high capacity density network inside a stadium can share LTE spectrum with the local macro eNodeBs in the area, as a secondary system using only a distributed Q-learning based DSA algorithm.

An Experimental Programme to Measure the Total Power Radiated by an Arcing Pantograph Using a Reverberation Chamber

Lan Ma$^{1,2}$, Andy Marvin$^{1,3}$, Yinghong Wen$^2$, Eva Karadimou$^3$, Rob Armstrong$^3$

$^1$Department of Electronics, University of York, York, UK
$^2$Beijing Jiaotong University, Beijing, China
$^3$York EMC Services, York, UK

Abstract. Pantograph arcing is the main electromagnetic interference source to radio communications (GSM-R) in an electric rail vehicle. A laboratory replica of pantograph arcing which can be operated in a reverberation chamber has been built. The replica has been verified by numerical models using the CONCEPT modelling program to show it is suitable to replace the real pantograph for investigating the radiation properties of pantograph arcing. The initial measurement results show that the reverberation chamber can catch the arcing signal well and does offer the possibility to measure the power radiated from the arcing pantograph. The average total radiated power in different resolution bandwidths (RBW), arc gap lengths and train speeds are measured according to IEC61000-4-21.
Key based fault monitoring in Network-on-Chip interconnection systems for enabling fault tolerant or fault evasive countermeasures.

Bharath Sudev

The Department of Computer Science
The University of York, United Kingdom YO10 5GH
bs638@york.ac.uk

Abstract. We present a key based fault monitoring technique for Network-on-Chip based systems to identify faulty connection lines so as to enable countermeasures to ensure reliable communication. The technique relies on error checking keys that would be send between routers at fixed intervals to identify faulty links and this would allow the system to take evasive countermeasures like adaptive routing or task migration or employ fault tolerant schemes. The routers equipped with this feature would have incrementing counters within them and when the counter overflows, a two flit wide key would be exchanged and then deflected back to the individual routers from the adjacent routers. This would help to verify sound operation of the connection lines in the finest detail. This work can complement dynamic fault tolerant/evasive approaches which usually deal with solely computational issues. This poster will present the key based evaluation system and present details on how the faulty connection lines stuck at 0 or 1 can be identified.

Keywords. Fault detection, Network-on-Chip, Fault key
The JUNIPER Project

Neil Audsley, Andy Wellings, Ian Gray, Jamie Garside, and Yu Chan
Department of Computer Science, University of York
{neil.audsley, andy.wellings, ian.gray, jamie.garside, yc522}@york.ac.uk

Abstract. The efficient and real-time exploitation of large streaming data sources and stored data poses many questions regarding the underlying platform, including:

Performance - how can the potential performance of the platform be exploited effectively by arbitrary applications; Guarantees - how can the platform support guarantees regarding processing streaming data sources and accessing stored data; and Scalability - how can scalable platforms and applications be built.

The fundamental challenge addressed by the project is to enable application development using an industrial strength programming language that enables the necessary performance and performance guarantees required for real-time exploitation of large streaming data sources and stored data.

The project's vision is to create a Java Platform that can support a range of high-performance Intelligent Information Management application domains that seek real-time processing of streaming data, or real-time access to stored data. This will be achieved by developing Java and UML modelling technologies to provide:

Architectural Patterns - using predefined libraries and annotation technology to extend Java with new directives for exploiting streaming I/O and parallelism on high performance platforms; Virtual Machine Extensions - using class libraries to extend the JVM for scalable platforms; Java Acceleration - performance optimisation is achieved using Java JIT to Hardware (FPGA), especially to enable real-time processing of fast streaming data; Performance Guarantees - will be provided for common application real-time requirements; and Modelling - of persistence and real-time within UML / MARTE to enable effective development, code generation and capture of real-time system properties.

The project will use financial and web streaming case studies from industrial partners to provide industrial data and data volumes, and to evaluate the developed technologies.
Identity of Players in Digital Games

Amelia Jati Robert Jupit and Paul Cairns
Department of Computer Science, University of York
{ajrj500, paul.cairns}@york.ac.uk

**Abstract.** Games allow players to master new challenges, the freedom to choose their own path and connect with other characters, either played by other players or non-playable characters (NPCs). On the other hand, players can be different people, using different identities than who they are in the real world. Identification in games is recognised for its potential component in game experience. This has prompted many studies in the attempt to define players' identity in games from perspectives such as avatar customisation, gender representation, narratives and self-discrepancies, to name a few. However, these studies differed at varying degrees in their concepts of player's identification with the game and focused mostly in role-playing games. My work aims to investigate how players form their identity in the games that they play. In this study, grounded theory was used to investigate what players themselves mean by identity and how they form their identities in their favourite game across multiple genres. Meaningful choices is found to be the key when players form their identity in the games that they play. This can be seen in the four aspects to the meaningful choices that players make: character, goals, gameplay and completion. Identity of players were found not just in the characters that they play, but that their identity occurs throughout their gaming experience as every action and decision made in the game lends a hand in forming their own identity.

**Keywords:** identity, digital games, game experience, grounded theory
Efficient Hybrid Impulse Response Synthesis for Room Acoustic Modelling

Stephen Oxnard\(^1\) and Damian Murphy\(^2\)

\(^1\) Audio Lab, The University of York, York, UK, YO10 5DD
so523@york.ac.uk
WWW home page: http://www-users.york.ac.uk/~so523

\(^2\) Audio Lab, The University of York, York, UK, YO10 5DD
so523@york.ac.uk
WWW home page: http://www-users.york.ac.uk/~dtm3

Abstract. Room acoustic modelling is the process of digitally emulating sound propagation throughout an enclosed, or partially enclosed, environment. Ideally, an acoustic model will allow for the faithful recreation of important acoustic characteristics inherent to the space under investigation. As such, the primary applications of this process include: acoustic prediction as an aid to architectural design and acoustic consultancy; the creation of realistic and immersive reverberation effects for game audio and other creative sound design practices; the sonic reproduction of soundscapes that are conceptual, inaccessible or no longer in existence.

The key goal is to capture the room impulse response (RIR), or “acoustic fingerprint”, of the sound field which encapsulates all of the acoustical information about the space. Most acoustic modelling methods fall into one of two categories: Geometric (GA) and Numerical (NA). It is broadly acknowledged that GA methods are efficient but lack accuracy at low frequencies. This is due to the underlying assumptions common to all GA methods which do not facilitate the preservation of wave phenomena and, therefore, low frequency sound characteristics. Numerical methods, such as the Finite Difference Time Domain (FDTD) and Digital Waveguide Mesh (DWM) paradigms, provide a direct solution of the wave equation. Therefore, NA methods are extremely accurate and provide valid results for the full audio bandwidth. However, this accuracy comes at high computational cost. This research exposes a novel hybrid acoustic modelling system that amalgamates both GA and NA paradigms. An efficient GA ray tracer is applied to calculate the high frequency portion of the RIRs. The low frequency impulse response is approximated by a “Multiplane FDTD model” : a series of cross-sectional 2D FDTD planes which captures a subset of 3D sound wave propagation. Reducing the dimensionality of the FDTD scheme leads to a significant reduction in computational expense. The results presented in this work pertain to the comparison of simulated low frequency multiplane FDTD RIRs and 3D FDTD RIRs demonstrating good agreement between the acoustic parameters captured in each modelling case. In addition, 2D multiplane simulations are shown to be far more efficient than 3D FDTD modelling procedures as they achieve, on average, a 98% reduction in computation time and memory requirement.
DreamCloud: Dynamic Resource Allocation in Many-core Embedded and High Performance Systems

Leandro Soares Indrusiak, Neil Audsley, Piotr Dziurzanski, and Amit Kumar Singh
Department of Computer Science, University of York
{leandro.indrusiak, neil.audsley, piotr.dziurzanski, amit.singh}@york.ac.uk

Abstract. DreamCloud is a Specific Targeted Research Project (STREP) of the Seventh Framework Programme for research and technological development (FP7). The project is a collaboration amongst market leading industrial organisations who develop products and services that are utilized in advanced systems, leading software tools and technology companies, and research and development organisations that develop advanced computing systems technologies.

The main objective of the DreamCloud project is to enable dynamic resource allocation in many-core embedded and high performance systems while providing appropriate guarantees on performance and energy efficiency. As case-studies, the project aims to realize three platforms: Embedded Clouds, Micro Clouds, and High Performance Clouds. Due to inherent differences between these platforms and the variety of potential applications, a bunch of assorted resource allocation algorithms has to be proposed. These algorithms need to utilize concepts from diverse domains such as control theory, evolutionary computing, market, and swarm intelligence. In this poster, we focus on the capability and applicability of these algorithms to the platforms considered for case-studies and their general applicability. Some preliminary experimental results are provided to demonstrate the efficiency of algorithms when employing various concepts.
NOMAD
Non-Standard Operand Mechanisms Architecture Demonstrator

Christopher Crispin-Bailey, Savan Vachhani

Department of Computer Science, University of York
{christopher.crispin-bailey, sv591}@york.ac.uk

Abstract. Computer Architecture has evolved with time. Over the course of time architects have come up with processors which are faster, and power efficient. Indeed architectures have come a long way but still there is a need (If it can be improved any further?). For very long time scaling down technology had been easy and efficient approach for improvements. But now improvements are difficult then what it was in the past as ‘Free Lunch is over’. The biggest names in the PC Processor aren’t satisfied with the status quo. IBM\footnote{Patent : http://www.google.com/patents/US8344358} and Intel are thinking for new transistor to keep scaling compute power. AMD and NVIDIA are inching toward heterogeneous system architecture. We, at the University of York, are also not satisfied with status quo and investigating whether stack processors can be a solution\footnote{Christopher Crispin Bailey Weblink: http://bit.ly/Xu4EZE}.

NOMAD\footnote{Christopher Crispin Bailey Weblink for NOMAD: http://bit.ly/1AUEJY7} is very basic 32 bit stack processor currently an integer processor. NOMAD is trying to demonstrate the feasibility of a low-power low-footprint ‘Nomadic’ computational engine, which is capable of operating in scalar or superscalar configurations. It is capable of issuing maximum 4 instructions per clock cycle with peak issue rate of 1.6 billion instructions per second. Prototype has been coded in VHDL, built using Cadence tools, and experiments are being conducted to evaluate performance. Silicon test chip for NOMAD will be built on 65nm technology and expected to be developed in early 2015.

As a next step we would like to investigate the opportunities for application of NOMAD processor. Some initial ideas include developing high performance data diagnostic low power sensors.

Stack is considered one of the most basic and natural tool for well-structured code. Researchers working on stack processors believe that there is a huge dearth of research on this area. Additionally because of some influences, myths and misconceptions have developed about Stack Processors. So stack processors can be tiny hope for change but we find reasons to investigate very credible.

Keywords: Stack Processor, NOMAD, Computer Architecture, Stack Machine
Automatic Paraphrase Extraction for Modern Standard Arabic

Amal Alshahrani
School of Computer Science
University of Manchester
Manchester M13 9PL, UK
Amal.alshahrani@postgrad.manchester.ac.uk

Abstract. The richness of human language allows people to express the same idea in many different ways; they may use different words or phrases for expressing and describing the same concept, i.e., paraphrases. Paraphrases play an important role in the variety and complexity of natural language documents. Many natural language applications, such as Information Retrieval, Machine Translation, Question Answering, Text Summarization, or Information Extraction need to process paraphrased documents correctly. However, analysing these documents at semantic level is a difficult task, so we build a paraphrase database to find expressions with similar meaning. Our aim is to create a system for extracting paraphrases for Modern Standard Arabic (MSA) from newswires, which contain articles that are good source for finding paraphrases and hence are likely to contain sentences with the same meaning if they report the same event on the same day.

There are existing algorithms for extracting paraphrase (e.g. TF-IDF vector space, cosine similarity) which have been applied to English. However, the performance of these algorithms could be affected when they are applied to Modern Standard Arabic due to the complexity of the language such as Free Word Order, Zero copula, and Pro-dropping. These Complexities will affect the performance of these algorithms. Thus, if we can analyse how the existing algorithms for English fail for Arabic then we can find a solution for Arabic. Our initial focus is to find many candidate paraphrases by comparing sentences that are likely to have the same meaning and looking for matching fragments by using Dynamic Time Warping (DTW) alignment methods.
Tools that Encourage Migration to Cloud Computing –
A Comparison between Adopters and Non-Adopters

Mohammed Almanea
School of Computing Science, Newcastle University
Newcastle upon Tyne, United Kingdom
mohammed.almanea@ncl.ac.uk

Abstract. A survey questionnaire was conducted in October 2012
(https://www.surveymonkey.com/s/GCYQRRG) to gather information from partici-
pants across several sectors including IT, education, healthcare, government, tele-
communication and banks. The survey aimed to understand cloud computing adoption
drivers and constraints, and whether tools such as the CSA STAR registry, Cloud-
Trust Protocol and CloudAssurance has been used and helpful, and whether they will
be utilised in the future. These tools have been chosen as they provide best industry
standards for potential cloud customers. In this poster, a comparison between adopters
and non-adopters is conducted. The comparison will present the results that have been
obtained from the respondents. The results include the perceived level of usage, the
level of helpfulness, and the potential level of usage in the future. CSA STAR is the
most used tool from the non-adopter’s point of view, whereas the CloudTrust Proto-
col and CloudAssurance are the most-used tools on the adopter’s side. With regard
to the helpfulness aspect, CloudAssurance is the most helpful tool as far as non-
adopters are concerned, whereas CSA STAR has been more helpful than other to the
adopters. The respondents (adopters) have shown more interest in using CloudTrust
Protocol in the future. The non-adopters plan to use CSA STAR in the future. There
are several reasons for the respondents’ selection of tools. Some of them mentioned
that they are unfamiliar with the tools, and depend on other frameworks such as the
gCloud. In addition, they pointed to a lack of defined requirements, a lack of trust,
and not enough information being provided.
Diagnosing Parkinson’s Disease with Evolutionary Algorithms

Stuart Lacy
Department of Electronics, University of York
stuart.lacy@york.ac.uk

Abstract. The treatment of Parkinson’s Disease can prove extremely costly and as of yet there are no available cures. Rapid and accurate diagnoses are essential for managing the condition as best as possible, however even among trained medical professionals there still remains a relatively high misdiagnosis rate of 25%. Ensemble classification is a popular field of machine learning research which combines predictions from multiple individual classifiers into one overall output. It has been shown to generalise well to unseen data and has high performance in the presence of noise. The objectives of this work were (1) to ascertain whether biologically-inspired machine learning techniques could be employed to detect the presence of Parkinson’s Disease symptoms in a simple movement task, and (2), whether combining such classifiers into an ensemble improves the prediction accuracy.

Movement data was recorded from both Parkinson’s Disease patients and healthy control subjects during a simple finger tapping exercise from 2 different medical centres. Both neural networks and standard Genetic Programming expression trees were trained using an Evolutionary Algorithm to classify raw data as belonging to either a Parkinson’s Disease patient or a healthy control. Ensemble classifiers were formed from the evolved populations and were assessed on the same data patterns.

The resulting classifiers could successfully distinguish between healthy control subjects and Parkinson’s Disease patients with a high degree of accuracy. The ensembles demonstrated a significantly greater prediction accuracy than the individual members. Analysing data recorded from simple, objective, and non-invasive tasks with computational intelligence techniques has been demonstrated to be highly accurate. This could lead to more accurate early diagnoses of Parkinson’s Disease and other Neurodegenerative conditions and thus increase treatment options. The ability of ensembles to bring together diverse clinical study data has been highlighted.
Development of an efficient model to increase public awareness on air pollution

Gabor Makrai¹, Prof. Peter Cowling², Steve Cinderby³, Stephen Remde⁴

¹ YCCSA, University of York, York, YO10 5GE, United Kingdom
gabor.makrai@york.ac.uk
² peter.cowling@york.ac.uk
³ steve.cinderby@york.ac.uk
⁴ stephen.rende@york.ac.uk

Abstract. According to the World Health Organisation⁴, while in 1970 36% of the population lived in urban area, in 2010, the ratio just increased over 50%. The process itself drives us to increased growth in cities and managing this process is getting more important than ever was before. According the studies, air pollution in urban area is mostly generated by traffic⁵. Increased amount of population living in urban area simply cause larger traffic volumes, but the urbanization process comes with increased amount of constructions to improve cities capacity for handling the increased amount of population.

It is clear that air pollution depends on technology however, it does depend on human activities controlled by our behaviour⁶. One way to reduce the pollution (through human behaviour) can be teaching the local population about the local environment and allow them to understand the environment where they live better.

An application designed to educate its users about their local environment and the pollution in that area can help to improve the public awareness on local pollution behaviour such as the sources of the pollution as well as the main factors which controls its dispersion.

Our research looks at the development of an computationally efficient way to model the pollution in urban environment. Combination of an urban traffic model⁷ and an air dispersion pollution model⁸ will be implemented in the way that it will be able to run on platforms with low computational power (such as smartphones).

Result of our research is going to be implemented into a computer game where users also can investigate the output of the local authorities’ decisions (such as road closures, speed limitations and implementation of different environmental policies).

⁶ Kollmuss, A., Agyeman, J. (2002). Mind the gap: why do people act environmentally and what are the barriers to pro-environmental behavior?
Accelerating the Linux Virtual File System

Russell Joyce
Department of Computer Science, University of York, UK
russell.joyce@york.ac.uk

Abstract. With the constant increase in the general speed of computers, traditional models of operating system kernels are increasingly becoming a limiting factor in computation time, due to the large overheads involved in system calls. Operations such as streaming large amounts of data from a network to a disk can cause high CPU load, especially if data is also being manipulated in real time as it is received, despite being relatively straightforward in terms of computation.

It is proposed that portions of an operating system may be accelerated by reimplementing them in hardware through the use of an FPGA. Specifically, the file system may be implemented in hardware to allow for accelerated functionality, as well as direct connection to other devices such as network controllers and application accelerators, without constant control from a CPU. This could improve performance for applications that stream large sets of data to and from non-volatile storage, especially in embedded systems with limited CPU resources.

This poster examines the layers of the Linux Virtual File System (VFS) and the operating system calls required for basic file operations, while profiling the kernel code with benchmarks to discover software overheads. The possibility of accelerating portions of the VFS, as well as the concrete file systems and hardware interfaces below it, is investigated.
Monitoring Water Pollution with a Heterogeneous Robotic Collective

Mayank Parmar, Steven Johnson and Jon Timmis

Department of Electronics, University of York, York YO10 5DD, UK
{mayank.parmar, steven.johnson, jon.timmis}@york.ac.uk

Abstract. Increasing levels of pollution in bodies of water poses a significant risk to human and animal health. The monitoring of rivers and lakes is a challenging task due, in part, to inaccessibility of the areas, combined with currents and tides that disperse pollutants. Current solutions like fixed location devices, such as sensor networks, are unable to predict the dispersal pattern of pollutants and also require knowledge of the location of pollutant for placing the monitoring device. Our work considers the use of a collective robotic monitoring system, that combines a variety of robotic platforms. These robots work together to identify, and then track pollutants in a water body. Such a system provides a decentralised and autonomous approach. By means of cooperation, the collective, or swarm, will work as a group, exploiting water monitoring devices on different robots in the collective. Monitoring data is combined with mapping data to allow for the creation of a pollutant map and also drives adaptivity of the collective to investigate areas of higher pollution. We aim to develop a heterogeneous collective system which allow robots to perform tasks independently, but also provide the flexibility to self-organise in to a collective, if further exploration, or specialist monitoring devices, are required.

In our proposed system, surface vehicles (catamarans) collects water samples and performs analysis of the collected sample. The water can be stored in one of a number of tanks for further investigation in the lab. A variety of sensors are used to navigate and control the catamarans, such as LIDAR. When combined with GPS, accurate pollutant-map can be generated. A mobile phone acts as the main processing power for the control system and provides access to phone sensors which gathers details about orientation and inertial effects. It also provides redundancy in communication. The multi-rotor aerial robot will act as a "virtual sensor" taking inspiration from work Pinto et al. in [1] and, in addition, will be used to perform visual monitoring and air quality checking.

Acknowledgements: This work is funded by the European Commission, Marie Curie Initial Training Network CAPACITIE Project. Project Reference: 608014 http://capacitie.eu/

References

Compact Low Phase Noise Oscillator at 3.8 GHz

Pratik D Deshpande, Jeremy Everard

Department of Electronics, University of York, York, YO10 5DD, United Kingdom
pd617@york.ac.uk

Abstract. This poster describes the theory and design of a compact low noise dielectric resonator oscillator operating at 3.8GHz. This oscillator demonstrates a phase noise performance of $-150 \text{ dBc/Hz}$ at 10 kHz offset with a tuning range of 200 kHz and vibration sensitivity between $10^{-9}$ and $10^{-8} \text{ g}^{-1}$. The power supply requirements are 6V without a regulator or 8V regulated at 160mA. The box dimensions are 11 x 11 x 5 cm. The oscillator is based on a feedback configuration consisting of a medium power amplifier, multi-section single layer 10dB output coupler, tunable electronic phase shifter, fixed phase shifter and a dielectric resonator with printed probes. Printed probes are used for coupling to the resonator which produces an unloaded Q of 19,000. Unloaded Qs up to 30,000 can be obtained for less robust configurations. Finally, vibration measurements were done at spot frequencies using a loud speaker and DC vibration motor at the University of York and full vibration measurements were performed at Selex-ES. The vibration sensitivity varied from $10^{-8}$ to $10^{-7} \text{ g}$ depending on the axes. A more modular but compact design is currently under investigation where the individual elements can be measured separately and then linked directly with each other to form the oscillator. A manual tuning phase shifter has also been designed so that the coaxial cables which are the sensitive to vibrations can be removed. The open loop phase shift is required to be accurate to $\leq 1 \text{ mm} (10^{9}/\text{mm})$ to achieve the required phase noise performance. Improvements in phase noise performance of around 8dB are predicted and expected.

Keywords. Low Noise Oscillators, Phase Noise, Vibration Measurements

Acknowledgements. We wish to thank Neil Raphael and Robin Collings for help with the vibration measurements and John Robinson for help with the project. The work was managed within the framework of the ITP SIMCLAIRS program. France, United Kingdom and Sweden mandated the European Defence Agency (EDA) to contract the Project with a Consortium composed of THALES SYSTEMES AEROPORTES France, acting as the Consortium Leader, Selex ES Ltd, THALES UK Ltd and SAAB AB.