TUNA : Final Report

Background and Context

TUNA ("Theory Underpinning Nanotech Assemblers") was a 2 year EPSRC-funded study into the feasibility of developing networks of nanites that behave safely, and into the computational languages, tools and techniques necessary to do so. Nanites are novel devices that operate on the nanoscopic level to cause macroscopic effects; nanotech assemblers are nanites that build macroscopic artefacts as an emergent property of their individual low level behaviours.

TUNA brought together a consortium of seven investigators from three institutions to perform this work. To focus the research, a specific case study was chosen from the field of nanomedicine: the process of artificial *haemostasis* (the staunching of bleeding) assisted by mechanical platelets. This case study was used as the basis for investigating the modelling languages, simulation tools and techniques, the structure of safety-case arguments, and emergent process design architectures. (Professor Susanna Hourani of the School of Biomedical and Molecular Sciences, Surrey, kindly helped educate the TUNA team in aspects of platelets and the blood clotting process.)

The £60k funding was used to purchase equipment, to perform and evaluate simulation experiments, and to support the investigators' travel, workshop, and conference expenses for meetings and dissemination. (This was a pre-FEC project: no investigator time was funded.) In addition to the investigators (Cavalcanti, Polack, Stepney, Woodcock from York; Welch from Kent; Schneider, Treharne from Surrey), other researchers (Barnes from Kent), PhD students (Turner from York; Jacobsen, Sampson from Kent), and a student intern (Ritson from Kent) also contributed to the research.

Achievements

The £60k TUNA feasibility study has resulted in: new formal methods theory, novel and fast concurrency mechanisms, novel massively parallel simulation demonstrators, 27 refereed publications, follow-on PhD and Masters research, and a £1.3M EPSRC funded follow-on project.

Formal modelling languages

Two approaches were considered for abstract modelling of nanite behaviours: *Circus* (a combination of CSP and Z), and CSP||B (a combination of CSP and B).

The modelling started from a CSP model of a cellular automaton that supports particles (abstract platelets) moving along a one-dimensional line. The CSP channels linking adjacent sites are used to control and synchronise the platelet motions. The model was extended to allow particles to have state ("smooth" and "sticky"), corresponding to inactive and active platelets), and to allow the sticky particles to clump. The model was gradually increased in sophistication, adding chemicals and diffusion. The final model takes advantage of CSP and B, in the style of CSP||B.

One surprising outcome of this work was just how hard it was to develop such models in CSP (even though the investigators involved are experts). The model was analysed with FDR and Probe, which helped to identify problems and solutions. The process of building models has pushed the theory forward in interesting ways, for example, it resulted in the definition of a *layered architecture* for this style of CSP model [12]. It also highlighted the need for mobility in languages such as *Circus* and CSP||B, and this is now being addressed.

Emergent property design architecture

Other approaches to modelling were also investigated, since it was felt that the "change of language" often present in the description of emergent systems might make classic refinement a difficult approach to follow [8]. In particular, the following question was tackled: how can one start with a specification of an emergent system or global property (such as a blood clot, or other emergent artefact desired of nano-assembly), and go from there to a design of the lower level agents with only local properties (such as platelets, or the nanites and their construction programs)? This led to the development of a layered "rule migration" architecture that supports system specification at high levels, or multiple levels, of emergence, then "migrates" the behaviour down to the lowest level necessary for implementation [9][16][17][18][19].

Argumentation structures

One feature of critical systems is that they must not only be correct, they must be seen to be correct: they need assurance arguments. Part of the TUNA work involved investigating what could be learned from conventional critical systems assurance arguments, such as Goal Structured Notation, and from work on emergent properties in macroscopic Systems of Systems. In particular, the role and validity of simulations, and in particular simulations of an emergent system's environment, as part of the assurance process, has been highlighted as a topic in need of further study. This has led to a Masters project developing a HAZOP (Hazard and Operability) based argumentation structure for nanites assemblers (ongoing).

Simulation system

occam-π is a concurrent programming language that builds on the CSP formalism, with extensions inspired by the π-calculus to add mobility. It has a very efficient native Linux implementation. The TUNA-grid infrastructure was used for two main tasks: to develop a multi-CPU version of occam-π to support scalable simulations, and to develop, run, and evaluate demonstration simulations distributed across the TUNA-grid [10][24].

Carl Ritson (student intern at Kent) used the TUNA grid to implement a distributed occam-π simulation of a three-layer model of environment, platelets, and clots following the architecture developed in the project. Each node supports a few million processes, limited only by RAM. Inter-node communication bandwidth and latency were sufficiently good to demonstrate reasonable and useful speedup factors from going parallel. These simulations, including a 36pp technical report describing the work, are available online at https://www.cs.kent.ac.uk/research/groups/sys/wiki/3D_Blood_Clotting

Implementation of the formal models grew to support three dimensions, mobile processes (for the platelets, chemical factors and blood clots), dynamic process creation, dynamic network topologies, distribution over PC clusters, 3D-visualisation and user interaction. We have reached the stage where simple *in silico* experiments can be performed – for example, to demonstrate the fine tolerance of platelet density of the bloodstream (too low or too high and the clotting response fails).

occam-π extensions

The simulation work, and the modelling work, exercised the capabilities of the various languages, and suggested extra programming language constructs, which have now been added to occam-π. These facilities, such as various multi-way (barrier) synchronisation protocols, including an efficient algorithm for resolving external choice over barriers, in principle can be programmed by a user, but they are complex and mistakes are easily made. Identifying these commonly used patterns [3][21][22], however, and adding them as specific high level language constructs, with a rigorous CSP semantics, enables the user to write and maintain occam-π systems at a higher level of abstraction. Specific extensions include mobile processes, channels and barriers [20][21]. The work forced the development of formal CSP models of these, which both enables model checking with standard CSP tools, and also documents low-level operational requirements for relevant aspects of the compiler and run-time kernel.

Other spinoffs

Barnes has been using the insights generated to develop a CSP compiler, with many of the restrictions of classical occam removed [2]. This compiler currently supports a subset of CSP, and is being extended to cover (almost) all of the algebra. This compiler will also support future developments of occam-π (which runs with the same run-time kernel used for CSP).

Project Plan Review

All tasks in the original project plan were covered, some in less detail than had been hoped, some in a great deal more detail.

Safety argumentation and reasoning about emergence were covered in outline, but have not yet resulted in formalised frameworks. However, this work is still ongoing, and enough has been determined about the feasibility of this approach to form the basis of a future project proposal.

The formal modelling was completed in depth, but not in the direction anticipated. The link between *Circus* and CSP||B was not established in the manner hoped. Instead, a layered architecture for CSP models was developed, and an appreciation for the need for mobility constructs in both formalisms was gained. The translation to *Circus* is now being tackled as a Masters project.

The simulation work was the most successful, far exceeding the original expectations and plans (partly due to the enthusiastic involvement of a further lecturer, and several students). The TUNA-grid performed extremely well, demonstrating scalability of simulations over tens of processors and tens of millions of processes.

Most gratifying was the interaction between the formal methods work and the simulation. Basing both aspects on a common underlying formalism, CSP, had benefits. The needs of the simulations to support the formal models drove developments in occam-π, and the extra mobility features in occam-π drove a recognition of their value in the formalism.

Research Impact

The project was a modestly-funded feasibility study: its whole purpose was to assess directions for future research. To this end it has been highly successful, leading to several further research activities (see later). It has resulted in the investigators making new interdisciplinary contacts, for example, part of the York team are now in discussions with the Biology Department in Abertay, about using the TUNA simulation techniques on complex ecological systems involving millions of organisms at multiple scales.

TUNA work has gained a high profile, having been disseminated widely, at a range of international conferences (covering formal methods, parallel computer architectures, and emergent systems), and in international journals (see conference and publication lists later). The TUNA PI was invited to give a presentation and arrange a special session on the TUNA work for the 11th IEEE International Conference on Engineering of Complex Computer Systems (*ICECCS'06*), in Stanford USA; this proved highly successful, and the organisers repeated their invitation for the following year's event.

Explanation of Expenditure

TUNA represents exceptional value for money: £60k, to allow the investigators to meet and work together, and to provide some kit to run experiments, has provided enough material and understanding to form a well-founded basis for several future research proposals.

TUNA grid

Approximately half the funding (~£26k) was spent of the "TUNA grid", an array of cheap Linux boxes supporting a native occam-π environment. The original proposal was costed for nine boxes connected by a 1 Gigabit Ethernet. By the time the project was funded, the falling price of hardware meant that the requested resources could be used to purchase 40 boxes, with 32 forming a TUNA-grid at Kent, and eight a mini-grid at York. Some money unspent from travel (~4k) was reallocated to pay for a student intern over the 2006 summer at Kent, to develop more substantial simulations running on the TUNA grid.

Project meetings

Approximately £10k was spent on investigator expenses and travel to attend TUNA meetings and dedicated project technical workshops. Full project meetings were held quarterly, rotating around the three sites, comprising planning sessions, presentations, and technical discussions. More focussed smaller one and two day technical workshops were held on an ad hoc basis (working out at approximately quarterly), for more detailed technical discussions, and formal modelling sessions.

Workshops and conferences

The remaining ~£20k was spent on travel, accommodation, and conference registration fees for disseminating TUNA work, and meeting external people working in related areas. The following conferences were attended by the named investigators (italics indicate attendees were funded specifically by TUNA).

1. ***ZB2005***, Guildford UK, April 2005 (*Cavalcanti*, *Schneider*, *Stepney*, Polack) – presentation at the associated REFINE 2005 workshop [8]
2. European Conference on Mathematical and Theoretical Biology (***ECMTB05***), Dresden Germany, July 2005 (*Stepney*) – invited presentation
3. 8th European Conference on Artificial Life (***ECAL'05***), Canterbury UK, Sept 2005 (*Polack*, *Stepney*, Welch) – workshop paper [17] and conference paper [9] on emergent architecture
4. Communicating Process Architectures (***CPA 2005***) Amsterdam Netherlands, Sept 2005 (*Welch*) – several occam-π presentations [1][4][5][10][21]
5. 4th International Conference on Unconventional Computation (***UC'05***), Seville Spain, October 2005 (*Polack*, *Stepney*)
6. 5th International Conference on Integrated Formal Methods (***IFM 2005***), Eindhoven Netherlands, November 2005 (*Schneider*, *Treharne*) – presentation on CSP||B with best paper award [11]
7. *Complexity in Design and Engineering*, Glasgow UK, March 2005 (Polack) – TUNA poster http://www.dcs.gla.ac.uk/~johnson/complexity/
8. *Rigorous Methods for Software Construction and Analysis*, Dagstuhl seminar 06191, May 2006, Germany (*Schneider*, *Treharne*) – presentation of the *IFM'05* TUNA work
9. 11th IEEE International Conference on Engineering of Complex Computer Systems (***ICECCS'06***), Stanford USA, Aug 2006 (*Barnes*, *Cavalcanti*, *Stepney*, *Woodcock*) – keynote and invited session on TUNA [12][16][22]
10. Formal Methods 2006 (***FM'06***), Aug 2006, Hamilton, Canada (*Schneider*, *Treharne*, Cavalcanti, Woodcock) – half day tutorial on CSP||B, including a session explicitly on TUNA's one dimensional CSP||B platelet model. Slides of that session 5 (and all sessions) available from www.csp-b.org, under 'supporting materials'. Also paper [6]

11. Communicating Process Architectures (***CPA 2006***) Amsterdam, The Netherlands, Sept 2006 (*Barnes*, *CR*, *Welch*) – occam-π presentations [2][13][23]

Further Research Activities

PhD projects

The TUNA work has fed directly into PhD projects. The "rule migration" emergence architecture model is informing Turner's thesis (at York). Various occam-π developments are informing the research of students at Kent (Sampson, Jacobsen, Dimmich, Schweigler). Further PhD student projects at Kent (Ritson, Brown, Bonicci) have started to extend areas of this work.

Developing new research proposals

Further responsive mode proposals are being prepared, based on

- use of the TUNA-grid simulation architecture for a range of complex emergent systems
- emergent property safety argumentation approaches
- approaches to validating simulations, including simulations of environmental properties
- extensions of CSP||B and *Circus* to handle mobility

CoSMoS

As a direct result of the TUNA project, CoSMoS ("Complex Systems Modelling and Simulation infrastructure") was submitted to EPSRC under their *Fundamentals of Complexity Science* call in November 2006. This 4 year, 2 RA, 6 RS £1.3M project, at York and Kent, is to build on the massively parallel simulation architecture developed and prototyped under TUNA, to provide generic modelling tools and simulation techniques for complex systems, to support the modelling, analysis and prediction of complex systems, and to help design and validate complex systems. The CoSMoS proposal was successful: the funding decision was made in March 2007.

Dissemination: Consolidated Publication list

Papers and posters have been published on all aspects of the TUNA work. Developments and dissemination of occam-π have been driven by the demands of the TUNA project, including several "fringe" presentations at CPA 2006. Several papers are concerned with developing the CSP||B approach, which underpins the CSP||B models developed in TUNA. Two papers are on a technique to translate *Circus* to Java using the JCSP library that implements most of the occam-π concurrency model: they are contributions to bridge the gap between *Circus* and the Kent group's work on occam-π. occam-π is one of the featured topics in the Non-classical computation Grand Challenge papers.

1. **F. R. M. Barnes**. Interfacing C and occam-π. In *CPA 2005*, pp249–260. IOS Press, 2005
2. **F. R. M. Barnes**. Compiling CSP. In *CPA 2006*, pp 377–388. IOS Press, 2006
3. **F. R. M. Barnes**, **P. H. Welch**, **A. T. Sampson**. Barrier synchronisation for occam-π. *Proc. 2005 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'05), Las Vegas USA, June 2005*. pp173–179. CSREA Press, 2005
4. J. F. Broenink, H. W. Roebbers, J. P. E. Sunter, **P. H. Welch**, D. C. Wood, eds, *Communicating Process Architectures (CPA 2005) Amsterdam, The Netherlands, September 2005*. Volume 63 of Concurrent Systems Engineering Series. IOS Press, 2006
5. D. J. Dimmich, **C. L. Jacobsen**. A Foreign Function Interface Generator for occam-π. *CPA 2005*, pp235–248. IOS Press, 2005
6. A. F. Freitas, **A. L. C. Cavalcanti**. Automatic Translation from *Circus* to Java. *FM'06*, LNCS **4085**:115–130. Springer, 2006

7. M. V. M. Oliveira, **A. L. C. Cavalcanti**, **J. C. P. Woodcock**. Formal development of industrial-scale systems. *Innovations in Systems and Software Engineering*, **1**(2):126–147, 2005
8. **F. Polack**, **S. Stepney**. Emergent Properties do not Refine. *REFINE 2005, Surrey, UK, April 2005*. *ENTCS* **137**(2):163–181, 2005
9. **F. Polack**, **S. Stepney**, **H. Turner**, **P. Welch**, **F. Barnes**. An Architecture for Modelling Emergence in CA-Like Systems. *ECAL 2005*, LNAI **3630**:427-436 Springer 2005
10. **A. T. Sampson**, **P. H. Welch**, **F. R. M. Barnes**. Lazy Cellular Automata with Communicating Processes. *CPA 2005*, pp165–175, IOS Press 2005
11. **S. Schneider**, **H. Treharne**, N. Evans. Chunks: component verification in CSP||B. *IFM'05*. LNCS **3771**:89–108, Springer, 2005 [best paper award]
12. **S. Schneider**, **A. Cavalcanti**, **H. Treharne**, **J. Woodcock**. A layered behavioural model of platelets. *ICECCS 2006*. IEEE 2006 [TUNA special session]
13. M. Schweigler, **A. Sampson**. pony - the occam-π Network Environment. In *CPA 2006*, pp 77–108. IOS Press, 2006
14. **S. Stepney**, S. L. Braunstein, J. A. Clark, A. Tyrrell, A. Adamatzky, R. E. Smith, T. Addis, C. Johnson, J. Timmis, **P. Welch**, R. Milner, D. Partridge. Journeys in non-classical computation I: A grand challenge for computing research. *International Journal of Parallel, Emergent and Distributed Systems*, **20**(1): 5–19, April 2006
15. **S. Stepney**, S. L. Braunstein, J. A. Clark, A. Tyrrell, A. Adamatzky, R. E. Smith, T. Addis, C. Johnson, J. Timmis, **P. Welch**, R. Milner, D. Partridge. Journeys in non-classical computation II: Initial journeys and waypoints. *International Journal of Parallel, Emergent and Distributed Systems*, **21**(2):97–125, April 2006
16. **S. Stepney**, **F. Polack**, **H. Turner**. Engineering Emergence. *ICECCS'06*. IEEE 2006 [invited keynote paper, TUNA special session]
17. **H. Turner**, **S. Stepney**. Rule Migration: Exploring a design framework for modelling emergence in CA-like systems. *ECAL Workshop on Unconventional Computing*, 2005
18. **H. Turner**, **S. Stepney**, **F. Polack**. A simulation environment for emergent properties (extended abstract). *ECCS'05, Paris France*, November 2005
19. **H. Turner**, **S. Stepney**, **F. Polack**. Rule Migration: Exploring a design framework for emergence. *Int. J. Unconventional Computing*. **3**(1): 2007
20. **P. H. Welch**, **F. R. M. Barnes**. Communicating mobile processes: introducing occam-π. In A. E. Abdallah, C. B. Jones, J. W. Sanders, eds, *25 Years of CSP*. LNCS 3525:175–210. Springer 2005
21. **P. H. Welch**, **F. R. M. Barnes**. Mobile Barriers for occam-pi: Semantics, Implementation and Application. *CPA 2005*, pp289–316, IOS Press 2005
22. **P. H. Welch**, **F. R. M. Barnes**, **F. A. C. Polack**. Communicating complex systems. *ICECCS 2006*. IEEE 2006 [TUNA special session]
23. **P. H. Welch**, J. Kerridge, **F. R. M. Barnes**, eds, *Communicating Process Architectures (CPA 2006) Amsterdam, The Netherlands, September 2006.* Volume 64 of Concurrent Systems Engineering Series. IOS Press, 2006
24. **P. H. Welch**, B. Vinter, **F. R. M. Barnes**. Initial experiences with occam-π simulations of blood clotting on the minimum intrusion grid. *Proc. 2005 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'05), Las Vegas USA, June 2005*. pp201–207. CSREA Press, 2005
25. Xinbei Tang, **Jim Woodcock**. Travelling Processes. *7th International Conference on the Mathematics of Program Construction (MPC 2004), Stirling UK, July 2004.* LNCS **3125**:381–399. Springer, 2004
26. Xinbei Tang, **Jim Woodcock**. Towards Mobile Processes in Unifying Theories. *2nd International Conference on Software Engineering and Formal Methods (SEFM 2004), Beijing China, September 2004*. pp44–53, IEEE, 2004
27. **Jim Woodcock**. Using *Circus* for Safety-critical Applications. *ENTCS* **95**:3–22, 2004