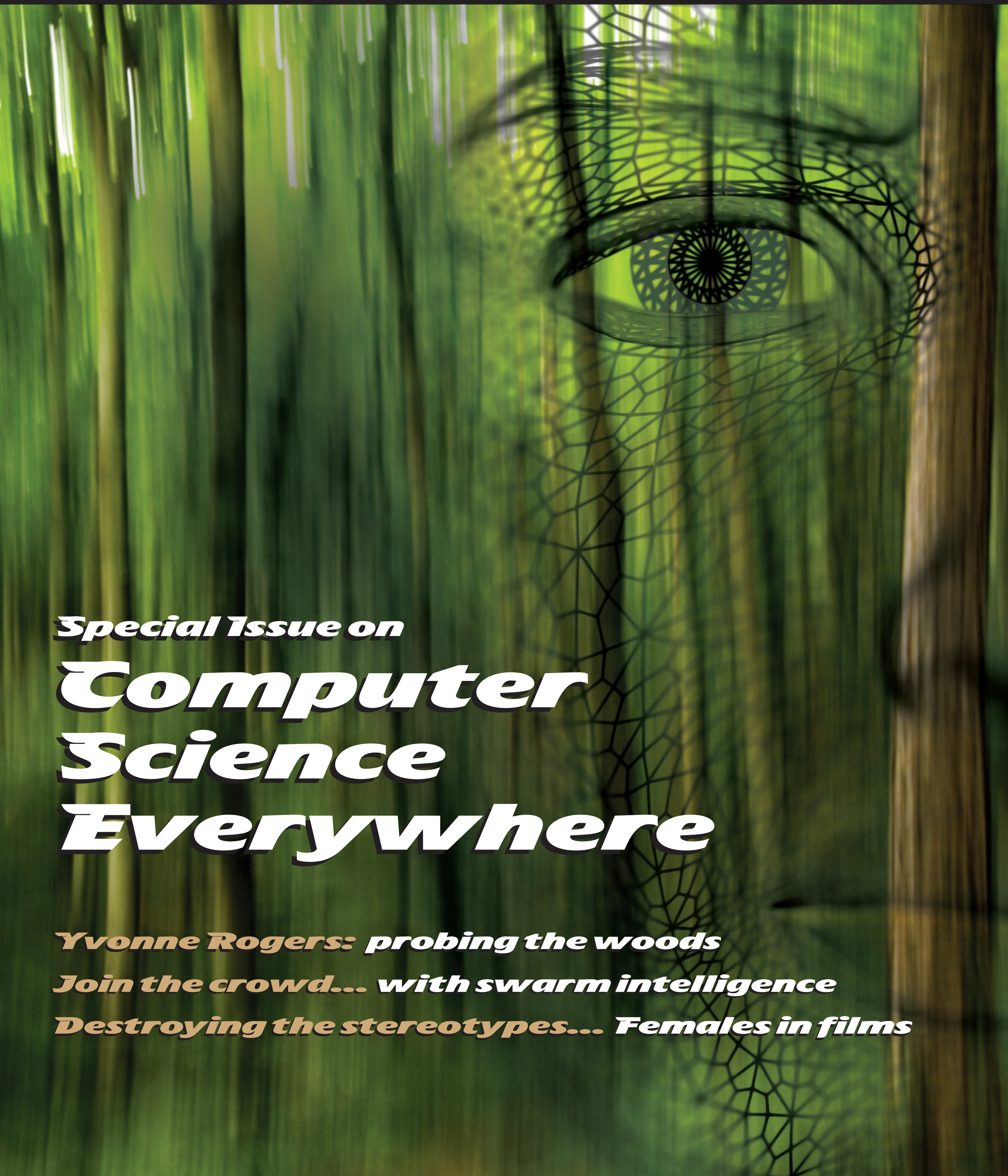
ISSN 1754-3657 (Print)
ISSN 1754-3665 (Online)

CS4FN

Computer Science for Fun

Issue 6



Special Issue on

Computer Science Everywhere

Yvonne Rogers: probing the woods

Join the crowd... with swarm intelligence

Destroying the stereotypes... Females in films

Special Issue on

Computer Science Everywhere

The computers have gone into hiding! Not long ago they were in your face saying "Look at me I'm a powerful computer". Now they are just as likely to be hiding in the crevices of your life. That's one goal of Computer Scientists – to help the computers hide!

You are probably so used to using your mobile phone you don't think of it as a computer, but that's what it is. Yours probably has as much computing power as the largest, most expensive computer did a generation ago.

Researchers want to create new technologies that are both mobile and fit seamlessly into our world. That is a goal of **ubiquitous computing**, or **ubicomp**, research: designing and creating computer technologies that are truly ubiquitous - present everywhere…but invisible.

This issue of cs4fn is on "Computer Science Everywhere" though not just in the UbiComp sense. Computer Science is everywhere in other ways too. It intertwines with other subjects in surprising ways. Even though computers have only existed for less than a century, the study of computation, which is what Computer Science is about, has been around for a thousand years or more…and has lead to cataclysmic changes to the history of civilisation. Computer Science is also at the vanguard of the exploration of space and the search for life.

Finally, Computer Science is for everybody: making a difference for the disadvantaged too, for example. It is not just for everyone in the sense of everyone being users, but also of everyone being able to contribute. Many people think it is only men who are interested in computers, but women are leading the way in many areas of research. We focus on the work of Yvonne Rogers of the Open University, Kirsten Cater of Bristol and pioneer Fran Allen of IBM. We also look at how SF films have reinforced the idea that it is male dominated and hear what some of the real IT Pros think. As a result this issue has been supported by Equalitec who work to promote IT for all.

As each new technology comes along it opens up whole new avenues for computer scientists to explore. IT now. Life tomorrow. Thanks to Computer Science.

In this issue:

I Predict a Riot

Professor Nigel Smart Head of the Department of Computer Science at Bristol University tells us about the world of mediascapes ...

It was Mark Weiser, chief scientist of Xerox PARC, and the visionary behind ubiquitous computing research, who set the ball rolling and the computers scurrying into the background with his dream of a future free from battles with ‘IT’:

“There is more information available at our fingertips during a walk in the woods than any computer system, yet people find a walk among the trees relaxing and computers frustrating. Machines that fit the human environment, instead of forcing humans to enter theirs, will make using a computer as refreshing as taking a walk in the woods.”

Researchers at Bristol University and the Open University together with collaborators like Hewlett Packard, the BBC and FutureLab have been taking small steps towards Weiser’s dream using mobile phones and personal digital assistants (PDAs) to give people ‘located’ experiences. These systems use natural human movements, like walking, to control applications automatically: the ‘computer’ changes behaviour depending on where you are.

One way a computer can keep track of location is the Global Positioning System (GPS) - used in cars for SatNav systems. The GPS satellites tell the mobile computer where it is. It then displays information appropriate to that location. Using GPS tracking to trigger the content you see means that you don’t have to be consciously interacting with the technology to control the experience. This kind of interaction reduces the need for complex user interfaces and makes the experience a lot more natural.

SatNav is the obvious application but there is a lot more potential than just SatNav for those innovative enough to see the possibilities. The Bristol team have called these types of experiences *‘mediascapes’*: a collection of sounds and images placed digitally in the landscape and accessed using mobile devices. They are exploring a whole bunch of different kinds.

The simplest mediascape application is to turn your mobile computer into a tourist guide. Let’s say Jo hasn’t been to Bristol before so she downloads a tourist guide to her GPS-enabled mobile phone. She starts walking around the harbourside, and when she approaches the SS Great Britain the mobile phone automatically starts to play audio files about the ship and shows video clips or images of the boat being launched. The effect is like audio guides in museums where you type in a number to get the specific content for that location, but using GPS means the system already knows where you are and where you have been. It is much closer to the way you’d interact with a human guide. Jo only has to walk around the ship naturally to trigger new content without having to interact with any complex interface or even speak.

The research team has gone much further and created a series of different kinds of applications each exploring different kinds of novel mediascape experiences – like taking part in a riot!

As Kirsten Cater of Bristol University who was one of the researchers behind that project, tells us: *“The idea behind Riot! 1831 was that when you walk into Queen Square you actually enter a virtual historical re-enactment of the 19th century riot that took place there and as you walk around the square you trigger different scenes”*.

Cititag, led by Open University researchers, by contrast explores ubicomp as a game – combining the physical and virtual worlds into a seamless gaming environment. Players roam their real home city searching for other players all of whom are in either ‘red’ or ‘green’. The aim is to tag players of the opposite colour before they tag you: *“Urban space becomes a playground and everyone is a suspect.”*

Most recently the Bristol team have been exploring how they can help everyone join in creating mediascapes. They have been trying the ideas out by letting students at Nailsea school design a mediascape of their yet to be built new school and so affect the design (see the webzine Magazine+).

In the future, location won’t be the only sensor that will be used in mediascapes. Other sensors like heart rate monitors, temperature and sound sensors could be used too (see Fizees, page 12). Here is one idea: mobile spy games might get you to run to a particular place but keeping your heart rate and sound level below a certain amount.

The real world has gone multimedia. Riots will never be the same again!

Future authoring tools may allow you to create a whole range of new sensor-based experiences with your friends wherever you like! You can already create your own location-based mediascapes. See the webzine, Magazine+ to find out more.

Someone to watch over me

They are everywhere! Cameras on lampposts, street corners, in shops or train stations. We sadly live in a world where crime is ever present and video surveillance plays a key part in helping the police keep us safe. How are computer scientists playing their part in changing the way we police our world today? Perhaps more importantly how far should we go?

Human smart vs Computer Smart

At present most of the surveillance cameras around us send their images to central control rooms where teams of observers keep an eye out for trouble. The problem is that as more and more pictures come in it becomes impossible for humans to keep track of all that's going on. Enter the smart camera. Smart because it has artificial intelligence. Even smarter because understanding what's going on in a moving picture, which is really, really easy for us, is really, really tough for a computer.

The human brain is estimated to use about half of all its computing power to understand the visual world. Take a simple example. Recognising a friend's face on either a dark and rainy day or a sunny day in a shadowy forest is easy for you. You don't really need to think about it. You take vision for granted (but that's because half your brain is working on it). But look at the information that a computer has to work with and these two cases are very different. The intensity values (that is the amount of light in the image) is very different in the two situations. The numbers making up the pixels of a digital image are completely different. In the forest the shadows change the light bits of a face to dark. The tip of the nose has say value 10 on the dark day and 100 on a sunny day. Somewhere in these two different sets of data is your friend's familiar face, but how can a computer find it? With current techniques it's still tough, and that's where computer science researchers come in to help find the answers. These difficult computational problems of duplicating human vision with a computer are just starting to be solved. Some fascinating advances have been made but there is still a long way to go.

There are lots of applications waiting once the problems of making computers 'see' are cracked. Of course even if a computer can recognise an expression, detect someone wandering around a car park in an unusual way or running from a station adding these clues together doesn't mean that we can decide the person's real intention, though we can alert someone to the fact that something different is happening. Computer Science researchers can provide the tools but it is the wider society that needs to decide whether these technologies are appropriate to use, and how they should be used. For better or worse we need to decide.

In the mood to face the computer

If you've ever found yourself pulling faces at your computer, becoming evermore frustrated as it fails to understand what you want it to do, then help from computer science researchers may be on its way. We use facial expressions to signal to others something about how we feel inside. It's one of the mechanisms we have evolved to let us live together in groups. Scientists are working on systems that teach a computer to recognise these facial expressions, and so get an idea of the moods or emotions those expressions might represent. A face is a very complex moving image. It's very bendy, but as we grow up we learn how to interpret these movements and turn them into useful measures of facial expression. We can now also teach a computer to recognise changes, as time progresses, of key parts of the face - known as 'spatio-temporal features'. By showing the computer a variety of expressive faces, and some clever algorithms, we can train it to tell the difference between, for example, a smile and a scowl.

The results can be used to help develop 'affective computing', that is computers that are aware of, respond to, and change according to our emotional state. For example, if you run into a problem while working with a particular piece of software, and start to look frustrated or angry, the computer will be able to recognise that emotion in your face, and automatically produce the right set of useful notes to help you. Such sensitive emotion-recognition software may appear on computers within the next few years.

Our plastic pal who is fun to be with?

But there are other applications for emotion-recognition software, for example in the development of 'synthetic companions' for the elderly. Robots can play a useful and valuable role in looking after the elderly or infirm – reminding them to take their medication, or to lock the front door or even perhaps acting as a type of pet you start to care for. But in order to build a relationship with an individual, to empathise with them, you have to be able to have a two-way communication. Expression recognition software would enable the robot to recognise a person's emotions and mood and respond appropriately – including reflecting the appropriate expression on their own face. You don't want a 'friend' who when you are unhappy appears to just smirk! Getting the emotions right is important in making the relationship between individual and carer much more socially realistic.

It's going to be important in the future that humans and robots 'talk the same language', even if no words are spoken.

Robots need emotions too!

Something in the way she moves

How do you tell if the person walking towards you is a man or woman? Getting it wrong can be incredibly embarrassing!

Hairstyle? Men have ponytails. Women have short hair.

Body shape? What if they are wearing baggy clothing?

Is there actually any feature that is a reliable way to tell?

Research by Kevin Shan at Queen Mary, University of London has shown that computers can be taught to identify whether a person is male or female, but in a surprising way. When we walk, our limbs act like pendulums, and the length and swing of the pendulum largely determines our individual pattern of movement. This swing is different for boys and girls; it's due to the difference in our body shapes and skeletons. By showing the computer examples of how lots of people walk the computer can be taught to recognise the difference.

All the clues point to...

To be even better at making the judgment we can combine this with computer facial analysis techniques that are trained to recognise the sorts of shapes of male and female faces. This bringing together of different clues to solve the problem is called 'data fusion'. It turns out it is a key idea in saving the lives of babies too *(see page 12)*.

So if you have ever been in that embarrassing situation when it dawns on you have been chatting up a woman not the man (or vice versa) you thought you were talking to, at least you know the emotional robots of the future won't have the same problem.

DATA FUSION

Data fusion can go even further combining these and other visual clues such as body temperature, the direction people are looking, or even how long a person has been standing still. From this set of clues a computer can, after it's been trained and knows the patterns to look for, decide if there is suspicious behaviour going on and alert the appropriate authorities.

Join the crowd... with swarm intelligence

Next time you are in a large crowd, have a look around you: all those people moving together, and mostly not bumping into each other. How does it happen? Flocks of birds and schools of fish are also examples of this "swarm intelligence". Swarming behaviour requires the individuals (birds, fish or people) to have a set of rules about how to interact with the individuals nearest to them. These so-called "local" rules are all that's needed to give rise to the overall or "global" behaviour of the swarm. We adjust our individual behaviour according to our current state but also according to the current state of those around us. If I want to turn left then I do it slowly so that others in the crowd can be aware of it and also start to turn. I know what I am doing and I know what they are doing. This is how information makes its way from the edges of the crowd to the centre and vice versa.

A swarm is born

The way a crowd or swarm interacts can be explained with simple maths. This maths is a way of approximating the complex psychological behaviour of all the individuals on the basis of local and global rules. Back in 1995 James Kennedy, a research psychologist, and Computer Scientist Russ Eberhart, having been inspired by the study of bird flocking behaviour by biologist Frank Heppner, realised that this swarm intelligence could be used to solve difficult computer problems. The result was a technique called PSO (particle swarm optimisation).

Travel broadens the mind

An optimisation problem is one where you have a range of options to choose from and you want to know the best solution. One of the classic problems is called 'The travelling salesperson' problem. You work for a company and have to deliver packages to say 12 towns. You look at the map. There are many different routes to take, but which is the one that will let you visit all 12 towns using the least petrol? Here the choices are the order in which you visit the towns, and the constraint is that you want to do the least driving. You could have a guess, select the towns in a random order and work out how far you'd have to travel to visit them in this order. You then try another route through all 12 and see if it takes less mileage, and so on. Phew! It could take a long time to work out all the possible routes for 12 towns to see which was best. Now imagine your company grows and you have to deliver to 120 towns or 1200 towns. You would spend all your time with the maps trying to come up with the cheapest solutions. There must be a better way? Well actually simple as this problem seems it's an example of a set of computational problems known as NP-Complete *(see Page 14)*, and it's not easy to solve! You need some guidance to help you through and that's where swarm optimisation come in. It's an example of a so-called metaheuristic algorithm: a sort of 'general rule of thumb' to help solve these types of problem. It won't always work unless you have infinite time but it's better than just trying random solutions. So how does swarm optimisation work here?

State space: the final frontier

First we need to turn the problem into something called a state space. You probably use state spaces all the time but just don't know it. Think about yourself. What are the characteristic you would use to tell people about your current state: age, height, weight and so on. You can identify yourself by a list of say 3 numbers – one for age, one for height, one for weight. It's not everything about you of course but it does define your state to some extent. Your numbers will be different to other people's numbers, if you take all the numbers for your friends you would have a state space. It would be a 3 dimensional space with axes: age, height and weight, and each person would be a point in that space at some coordinate (X, Y, Z).

So state spaces are just a way of representing the possible variations in some problem. With the 12 towns problem however you couldn't draw this space: it would be 12 dimensional! It would have one axis for each town, with the position on the axis an indication of where in the route it was. Every point in that space would be a different route through the 12 towns, so each point in the space would have coordinates (x1, x2, x3, … x11, x12). For each point there would also be a mileage associated with the route, and the task is to find the coordinate point (route) with the lowest mileage.

Swarming all over the map

Enter swarm optimisation. We create a set of 'particles' that will be like birds or fish, and will fly and swarm through our state space. Each particle starts with a random starting location in the state space and calculates the mileages involved for the coordinates (route) they are at. The particles remember (store) this coordinate and also the value of the mileage at that position. Each particle therefore has it's own known 'local' best value (where the lowest mileage was) but can compare this with other neighbouring particles to see if they have found any even better solutions. The particles then move onwards randomly in a direction that tends to move them towards their own local best and the best value found by their neighbours. The particles 'fly' around the state space in a swarm homing in on even better solutions until they all converge on the best they can find: the answer.

Where no particle has swarmed before

It may be that somewhere in a part of the space where no particle has been there is an even better solution, perhaps even the best solution possible. Our swarm will have missed it! That's why this algorithm is a heuristic, a best guess to a tough problem. We can always add some more particles at the start to fill more of the state space and reduce the chance of missing a good solution, but we can't ever be 100% sure.

Swarm optimisation has been applied to a whole range of tough problems in computing, electronic engineering, medicinal chemistry and economics. All it needs is for you to create an appropriate state space for your problem and let the particles fly and swarm to a good solution.

It's yet another example of clever computing based on behaviours found in the natural world. So next time you're in a crowd, look around and appreciate all that collective interacting swarm intelligence… but make sure you remember to watch where you are stepping.

The Dark History of Algorithms

Zin Derfoufi, a Computer Science undergraduate at Queen Mary, delves into some of the dark secrets of algorithms past.

Algorithms are used throughout modern life for the benefit of mankind whether as instructions in special programs to help disabled people, computer instructions in the cars we drive or the specific steps in any calculation. The technologies that they are employed in have helped save lives and also make our world more comfortable to live it. However, beneath all this lies a deep, dark, secret history of algorithms plagued with schemes, lies and deceit.

Algorithms have played a critical role in some of History's worst and most brutal plots even causing the downfall and rise of nations and monarchs. Ever since humans have been sent on secret missions, plotted to overthrow rulers or tried to keep the secrets of a civilisation unknown, nations and civilisations have been using encrypted messages and so have used algorithms. Such messages aim to carry sensitive information recorded in such a way that it can only make sense to the sender and recipient whilst appearing to be gibberish to anyone else. There are a whole variety of encryption methods that can be used and many people have created new ones for their own use: a risky business unless you are very good at it.

One example is the ‘Caesar Cipher’ which is named after Julius Caesar who used it to send secret messages to his generals. The algorithm was one where each letter was replaced by the third letter down in the alphabet so A became D, B became E, etc. Of course, it means that the recipient must know of the algorithm (sequence to use) to regenerate the original letters of the text otherwise it is useless. That is why a simple algorithm of “Move on 3 places in the alphabet” was used. It is an algorithm that is easy for the general to remember. With a plain English text there are around 400,000,000,000,000,000,000,000,000 different distinct arrangements of letters that could have been used! With that many possibilities it sounds secure. As you can imagine, this would cause any ambitious code-breaker many sleepless nights and even make them go bonkers!

It became so futile to try and break the code that people began to think such messages were divine!

But then something significant happened. In the 9th Century a Muslim, Arabic Scholar changed the face of cryptography forever. His name was Abu Yusuf Ya'qub ibn Ishaq Al-Kindi – better known to the West as Alkindous. Born in Kufa (Iraq) he went to study in the famous Dar al-Hikmah (house of wisdom) found in Baghdad. It was the centre for learning in its time which produced the likes of Al-Khwarzimi, the father of algebra - from whose name the word algorithm originates. It also produced many more scholars who have shaped the fields of engineering, mathematics, physics, medicine, astronomy, philosophy and every other major field of learning in some shape or form.

Frequency Analysis changed the course of history

Al-Kindi introduced the technique of code breaking that was later to be known as 'frequency analysis' in his book entitled: *‘A Manuscript on Deciphering Cryptographic Messages’*.

His idea was that to decrypt a message all we have to do is find out how frequent each letter is in both the sample and in the encrypted message and match the two. If E is most common in English, it is likely to be in the message too. Obviously common sense and a degree of judgement has to be used where letters have a similar degree of frequency. Although it is a lengthy process it certainly was the most efficient of its time and, most importantly, effective.

As decryption became possible, many plots were foiled changing the course of history. An example of this was how Mary Queen of Scots, a Catholic, plotted along with loyal Catholics to overthrow her cousin Queen Elizabeth I, a Protestant, and establish a Catholic England. The details of the plots carried through encrypted messages were intercepted and decoded. On 15 October 1586 Mary was on trial for treason. Her life had depended on whether one of her letters had been decrypted or not. In the end, she was found guilty and publicly beheaded for high treason. Walsingham, Elizabeth's spymaster, knew of Al-Kindi's approach.

Today encryption is a major part of our lives in the form of Internet security and banking. Learn the art and science of encryption and decryption and who knows, maybe some day you might succeed in devising a new uncrackable cipher or crack an existing banking one! Either way would be a path to riches! So if you thought that algorithms were a bore … it just got a whole lot more interesting.

Smart Translation

Computers don't speak English, or Urdu or Cantonese for that matter. They have their own special languages that human programmers have to learn if they want to create new applications. Even those programming languages aren't the language computers really speak. They only understand 1s and 0s. The programmers have to employ translators to convert what they say into Computerese (binary): just as if I wanted to speak with someone from Poland, I'd need a polish translator. Computer translators aren't called translators though. They are called 'compilers', and just as it might be a Pole who translated for me into Polish, compilers are special programs that can take text written in a programming language and convert it into binary.

The development of good compilers has been one of the most important advancements from the early years of computing and Fran Allen, one of the star researchers of computer giant, IBM was awarded the 'Turing Prize' for her contribution. That is the Computer Science equivalent of a Nobel Prize. Not bad given she only joined IBM to clear her student debts from University.

Fran was a pioneer with her groundbreaking work on 'optimizing compilers'. Translating isn't just about taking a word at a time and substituting each for the word in the new language. You get gibberish that way. The same goes for computer languages.

Things written in programming languages are not just any old text. They are instructions. You actually translate chunks of instructions together in one go. You also add a lot of detail to the program in the translation, filling in every step.

Suppose a Japanese tourist used an interpreter to ask me for directions of how to get to Sheffield from Leeds. I might explain it as "Follow the M1 South from Junction 43 to Junction 33". If the Japanese translator explained it as a compiler would they might actually say (in Japanese): Take the M1 South from Junction 43 as far as Junction 42, then follow the M1 South from Junction 42 as far as Junction 41, then follow … from Junction 34 as far as Junction 33. Computers actually need all the minute detail to follow the instructions.

The most important thing about computer instructions (i.e., programs) is usually how fast following them leads to the job getting done. Imagine I was on the Information desk at Heathrow airport and the tourist wanted to get to Sheffield. I've never done that journey. I do know how to get from Heathrow to Leeds as I've done it a lot. I've also gone from Leeds to Sheffield a lot, so I know that journey too. So the easiest way for me to give instructions for getting from London to Sheffield, without much thought and be sure it gets the tourist there might be to say:

Go from Heathrow to Leeds:

1. Take the M4 West to Junction 4B
2. Take the M25 clockwise to Junction 21
3. Take the M1 North to Leeds at Junction 43

Then go from Leeds to Sheffield

4. Take the M1 South to Sheffield at Junction 33

That is easy to write and made up of instructions I've written before perhaps. Programmers reuse instructions like this a lot – it both saves their time and reduces the chances of introducing mistakes into the instructions. That isn't the optimum way to do the journey of course. You pass the turn off for Sheffield on the way up. An optimizing compiler is an intelligent compiler. It looks for inefficiency and actually converts it into a shorter and faster set of instructions. The Japanese translator, if acting like an optimizing compiler, would actually remove the redundant instructions and simplify it to:

1. Take the M4 West to Junction 4B
2. Take the M25 clockwise to Junction 21
3. Take the M1 North to Sheffield Junction 33

Much faster! Much more intelligent! Happier tourists!

Next time you take the speed of your computer for granted, remember it is not just that fast because the hardware is quick, but because, thanks to people like Fran Allen, the compilers don't just do what the programmers tell them to do. They are far smarter than that.

If you go down to the woods today

What kind of emotion should computers evoke? Calm? Frustrating? Professor Yvonne Rogers tells us about her vision for the future.

Mark Weiser's dream of "Calm Computing" has led to lots of exciting research that has seen computers disappearing into the background *(see page 3)*. His vision was driven by a desire to remove the frustration of using computers but also the realization that the most profound technologies are the ones that you just don't notice. He wanted technology to actively remove frustrations from everyday life, not just the ones caused by computers.

No one argues that computers should be frustrating to use, but Yvonne Rogers has a different idea of what the new vision could be. Not calm. Anything but calm in fact (apart from frustrating of course). Not calm, but engaging and exciting!

Not calm, but engaging and exciting

Her vision of those tranquil woods is not relaxing but provocative and playful. To prove the point her team turned some real woods in Sussex into an 'Ambient Wood'. The ambient wood was an enhanced wood. When you entered it you took probes with you, that you could point

and poke with. They allowed you to take readings of different kinds in easy ways. Time hopping 'Periscopes' placed around the woods allowed you to see those patches of woodland at other times of the year. There was also a special woodland den where you could then see the bigger picture of the woods as all your readings were pulled together using computer visualisations.

Not only is the Ambient Wood technology visible and in your face but it makes the invisible side of the wood visible in a way that provokes questions about the wildlife. You notice more. You see more. You think more. A walk in the woods is no longer a passive experience but an active, playful one. Woods are the exciting places of childhood stories again but now there are even more things to explore.

"The most important thing the participants gained was a sense of wonderment at finding out all sorts of things and making connections through discovering aspects of the physical woodland (e.g., squirrel's droppings, blackberries, thistles)" — Yvonne Rogers

The idea behind the Ambient Wood, and similar ideas like Bristol's Savannah project where playing fields are turned into African Savannah (see the webzine Magazine+) is to revisit the original idea of computers but in a new context. Computers started as tools, and tools don't disappear, they extend our abilities. Tools originally extended our physical abilities – a hammer allows us to hit things harder, a pulley to lift heavier things. They make us more effective and allow us to do things a mere human couldn't do alone. Computer technology can do a similar thing but for the human intellect…if we design them well.

The Weiser dream is that technology invisibly watches the world and removes the obstacles in the way before you even notice them. It's a little like the way servants to the aristocracy were expected to always have everything just right but at the same time were not to be noticed by those they served. The way this is achieved is to have technology constantly monitoring, understanding what is going on and how it might affect us and then calmly fixing things. The problem is it needs really 'smart' technology – a high level of Artificial Intelligence to achieve and that so far has proved more difficult than anyone imagined. Our behaviour and desires are full of subtlety and much harder to read than was imagined. Even a super-intellect would probably keep getting it wrong.

There are also ethical problems. If we do ever achieve the dream of total calm we might not like it. It is very easy to be gung ho with technology and not realize the consequences. Calm computing needs monitors – the computer measuring everything it can so it has as much information as possible to make decisions from *(see Big Sister is Watching You, page 12)*.

A classic example of how this can lead to people rejecting technology intended to help is in a project to make a "smart" residential home for the elderly. The idea was that by wiring up the house to track the residents and monitor them the nurses would be able to provide much better care, and relatives be able to see how things were going. The place was filled with monitors. For example, sensors in the beds measured resident's weight while they slept. Each night the occupants weight could invisibly be taken and the nurses alerted of worrying weight loss over time. The smart beds could also detect tossing and turning so someone having bad nights could be helped. A smart house could use similar technology to help you or I have a good nights sleep and help us diet.

The problem was the beds could tell other things too: things that the occupants preferred to keep to themselves. Nocturnal visitors also showed up in the records. That's the problem if technology looks after us every second of the day, the records may give away to others far more than we are happy with.

Yvonne's vision is different. It is not that the computers try to second-guess everything but instead extend our abilities. It is quite easy for new technology to lead to our being poorer intellectually than we were. Calculators are a good example. Yes we can do more complex sums quickly now, but at the same time without a calculator many people can't do the sums at all. Our abilities have both improved and been damaged at the same time. That is what the probes do, allowing you to see the woods in a new way, but to use the information however you wish. Crucially the probes encourage imagination too.

The alternative to the smart house (or calculator) that pampers allowing your brain to stay in neutral, or the residential home that monitors you for the sake of the nurses and your relatives is one where the sensors are working for you. Where you are the one the bed reports to helping you to then make decisions about your health, or where the monitors you wear are part of a game that you play because its fun *(see Fizees page 12)*.

What next?

"I'd like to see kids discover new ways of probing their bodies to find out what makes them tick." — Yvonne Rogers

So if Yvonne has her way, you won't be heading for a soporific future while the computer deals with real life for you. Instead it will be a future where the computers are sparking your imagination, challenging you to think, filling you with delight...and where the woods come alive again just as they do in the storybooks.

For the full cs4fn interview with Yvonne see the cs4fn webzine Magazine+

Fizzees: the digital pet that cares for YOU

Who says computers can't be part of a healthy lifestyle? It's often assumed that spending too long playing games with computers has caused many of us to stop doing the daily exercise that helps keep us fit and healthy. The Fizzees project developed by Futurelab aims to change that, by giving kids a clever bit of wearable technology: a watch containing their own cute digital Tamogotchi-style pet. You can watch it grow and develop just like any pet, but the twist is that you need to do the work in the real world for your pet to prosper in its digital domain.

Watch your watch

The way your digital pet grows and develops depends on the physical activity you do. The watch device containing the pet is fitted with a monitor that measures your heart rate and your movement. A good bit of exercise means that your pulse increases and you move about a lot more, and this is converted into information that helps your pet grow. Using these devices people start to understand what it takes for a healthy lifestyle. The pet does best when its wearer undertakes the sorts of healthy activities health experts think are most suitable for their age group.

Putting the fun into healthy

This interactive device is an example of how a clever idea combined with wearable computing can lead to devices that can improve our lives and also be a lot of fun as well. Perhaps in the future we will all have wearable digital devices that help us have healthier lifestyles: digital pets that actually look after us.

Big Sister is watching

The word "Surveillance" can have sinister overtones. You're being watched. Everything you do. All the time...

There is at least one time in your life when you might have wanted such 24-hour coverage: if you were a premature baby in an incubator, it would be about life or death. Babies are wired up to lots of probes that sound an alarm if anything strange is detected. The trouble is the alarms go off all the time – even because a baby wriggles. For example, if a probe says the heart stopped instantly, the baby hasn't died. It just means a probe has become detached – babies' hearts just do not stop suddenly like that. Over 90% are false alarms making a baby unit a very noisy place, but more importantly it's possible real alarms could be missed.

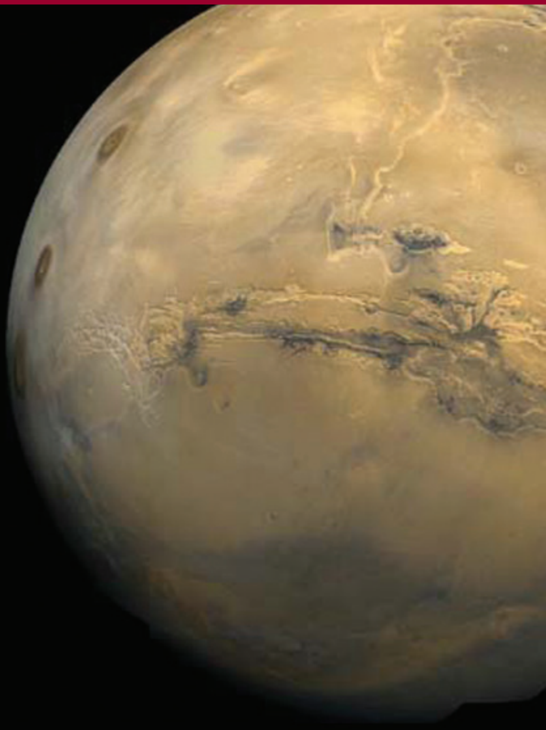
Enter some Computer Science from the University of Edinburgh. Professor Chris Williams and John Quinn of the School of Informatics, and Professor of Child Life and Health, Neil McIntosh with funding from the charity BLISS have developed a computer system that is capable of monitoring all the probes and only raising the alarm when the combined pattern of data suggests a problem.

Computer Scientists have devised clever ways for computers to look for patterns in data, and these methods can then be used to solve lots of different problems, including the baby monitor. Combining Computer Science methods with expert knowledge from Neil has resulted in a system that has far fewer false alarms – but doesn't miss the real problems.

Night, Night, Sleep tight!

Baby units could be much more peaceful in the future, thanks to the computer, always watching.

Life on Mars?



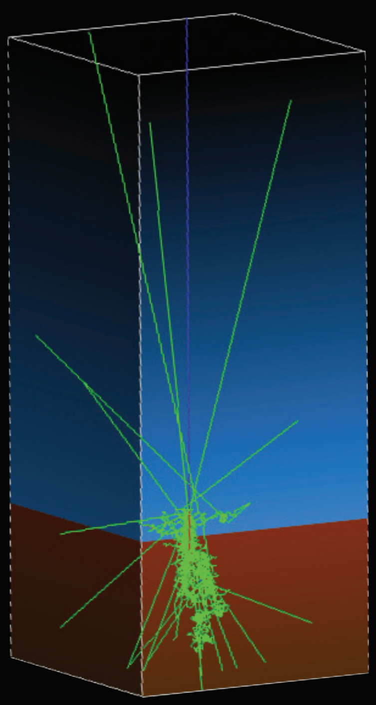
Lewis Dartnell, a PhD student at UCL and author of "Life in the Universe: A Beginner's Guide" tells us about his search for life…

I am searching for life on Mars. Now, before you dismiss me as a crank, let me explain...

I carry out research in the field of astrobiology: the study of the 'origin, evolution, distribution and future of life in the Universe'. Mars, our planetary next-door neighbour, has a good chance of harbouring life. The Martian environment billions of years ago was much like Earth's, with a thick warm atmosphere, seas and lakes of water on the surface, and probably lots of interesting organic chemistry going on. The problem is that Mars suffered a cataclysmic environmental collapse.

Most of Mars' atmosphere has blown into space and its surface is a dust-swept, freeze-dried desert. If life did evolve here long ago then it's coping with some pretty unpleasant conditions on the Martian surface right now. The ground is so cold that any bacterial life remaining must be in deep freeze, held dormant in the preserving ice.

Another hazard on the Martian surface is radiation from outer space. Cosmic rays are made up of particles accelerated to near the speed of light by solar flares and exploding stars throughout the galaxy. These energetic particles are extremely dangerous to organisms as they shatter the delicate molecules of life such as DNA and proteins. The Earth is shielded by a strong magnetic field and a nice thick atmosphere, but Mars has been laid bare to this onslaught of radiation. So the big question is, how long can frozen bacteria survive the radiation on Mars' surface before they're all killed off?



Mars suffered a cataclysmic environmental collapse

We have a good idea what the radiation environment is in space, through measuring it with space probes. No particle detectors have actually landed on Mars but calculating the radiation levels on, and beneath, the Martian surface is a perfect problem to attack with computer models. That is the essence of my PhD research so far: developing a computer program to handle all of the particle physics going on with this high-energy radiation, feeding in all the information about the properties of the Martian atmosphere and surface, deciding what data would be meaningful for the model to output, and then letting a network of computers churn through all the complicated maths to produce the desired results.

The image shows a visualisation of the set-up for my computer model. I have designed it with a block of Mars rock beneath a column of atmosphere, realistically layered so that the air gets denser nearer the ground. An energetic cosmic ray (in blue) can be seen approaching Mars from outer space, punching straight through the thin atmosphere and then triggering a great shotgun blast of radiation underground (the red and green cascade). I need information like how many radiation particles are created, of which kind, and what radiation dose they deposit within bacterial cells living at different depths underground. That allows me to work out how long cells might survive.

The particle physics going on inside these radiation cascades is enormously complicated, with millions of new particles being created from each original cosmic ray, of tens of different kinds, and all interacting in different ways. The program uses a common trick of computer models to work out what to do at each step in time it simulates. During every time-step each of these fast-moving particles can decay, move forward, or collide into something – interacting in many different ways. Physicists have worked out the probability of each happening. All the computer model needs to do to simulate every time-step then is pick a random number and check which of the possibilities this relates to. This is like rolling a large multi-sided dice, and looking the result up in a table of possible outcomes. It's called a 'Monte Carlo' simulation, after the famous gambling casino resort.

After I've modelled the complete cascades generated by large numbers of original cosmic ray particles and calculated the radiation dose at different depths beneath Mars' surface I can use these to approximate how long bacteria might survive at different depths.

The answer? Well the longest drill that space engineers have so far designed for a probe bound for Mars is 2m. Looking at that depth even the most radiation resistant bacteria known would be killed off in less than half a million years.

It looks like we're either going to have to drill deeper to find surviving life on Mars, or land our probes at special locations where life may have been brought to the surface very recently. Not bad from a computer model that simulates space radiation by rolling a few dice … several billion times...

Tantrix: P=NP

You can find computer science everywhere – even in popular Solitaire games and puzzles. Most people play games like Tetris, Battleships, Mastermind, Tantrix and Minesweeper at some point. In fact all these games have a link to one of the deepest, fundamental problems remaining in Computer Science. They are all linked to a famous equation that is to do with the ultimate limitations of computers.

The sciences have many iconic equations that represent something fundamental about the world. The most famous is of course Einstein's $E=mc^2$, which even non-scientists have heard of. The most famous equation in computer science is "P=NP". The only trouble is no one has yet proved whether it is true or not! There is even a million dollar prize up for grabs for anyone who does, not to mention great fame!

P=NP boils down to the difference between checking if someone's answer to a puzzle is correct, as against having to come up with the answer in the first place.

You are so NP!

Computer Scientists call problems where it is easy to check answers "NP problems".

Ones where it's also easy to come up with solutions are "P problems".

So P=NP, if it were true, would just mean that all problems that are easy to check are also easy to solve.

Let's take Tantrix rotation puzzles to see what it is all about. Tantrix is a popular domino-type game using coloured hexagonal pieces like the ones here. The idea of a Tantrix rotation puzzle is that you place some tiles randomly on the table in a connected pattern. You are then not allowed to move the position of any piece. All you can do is rotate them on the spot. The problem is to rotate the pieces so that all the coloured lines match where tiles meet – red to red lines, blue to blue lines and so on.

Have a go at the Tantrix rotation puzzle here before you read on.

Easy to check?

A solution is in the cs4fn webzine if you want to check it. In fact you can quickly check any claimed rotation puzzle 'solution'. All you do (the checking 'algorithm') is look at each tile in turn and check each of its edges does match the edge of the tile it touches, if any. If you find a tile edge that doesn't match then the 'solution' isn't a solution after all. How long would that take to check? With say a 10 piece puzzle there are 10 pieces to check each with 6 edges so in total that is 10x6 = 60 things to check. That wouldn't take too long. Even with 100 pieces it would be only 600 things to check – 10 minutes if you could check an edge a second. So Tantrix rotation puzzles are NP puzzles – they can be checked quickly.

But can you solve it?

The question is: "Are rotation puzzles P puzzles too?" Can they always be solved quickly if you or a computer were clever enough? You may have found that puzzle easy to solve. It is much harder to come up with a quick way that is guaranteed to solve any rotation puzzle I give you. One way would be to methodically work through every combination of tile rotations to see if it worked. That would take a long time though.

There are 6 positions for the first piece (ways to rotate it), but for each of those 6 positions the second piece could be in 6 positions too ... and so on for each other piece. Altogether for a 10-tile puzzle there are 6x6x6x6x6x6x6x6x6x6 (i.e., over 60 million) positions to check looking for a solution (and we might have to check them all). If you could check one position a second, it would take you around 700 days non-stop (no eating or sleeping). That is just for a 10-tile puzzle...now for a 100-tile puzzle – I'll leave you to work that out.

It is not what computer scientists call "quick".

How clever do you have to be?

If P=NP is true it would mean there is a quick way of solving all Tantrix rotation puzzles out there, if only someone were clever enough to think of it. If P=NP is false then it might just not be possible however clever you are. Trouble is no-one knows if it is true or not…

To find out more about Tantrix links to computer science and what are known as NP-complete problems, go to the cs4fn webzine Magazine+.

Optimization is one way to get good, if not perfect, solutions to hard problems

Carry on conjuring

You gather up the pile of cards from your last trick (perhaps the 21-card trick – see the cs4fn webzine Magazine+) after triumphantly revealing the card the volunteer was thinking of. You now show that you can not only read minds, but also see into the future.

First, you write a prediction on a piece of paper and seal it in an envelope so no one sees your prediction. You give it to a member of the audience to hold so that the sealed envelope remains in clear sight and cannot be tampered with.

Next you ask the spectator to cut about half the pack off the top. They decide how much, free choice. They are going to select a card from the top half of the pack that they just cut off but even they aren't going to know which one it will be.

They deal the first card face down on to the remnants of the pack, and the next card face up on the table, next card face down on the remnants of the pack, next face up on the table, and so on.

Once they have finished with the cards in their hands they start again, picking up the face up pack turning over and dealing the first face down on the pack remnants and the next face up, until all cards are dealt.

Again they pick up the face-up cards and deal in the same way. They continue doing this until they have exhausted the cards in their hand and there is only one left face up on the table.

You recap for the audience: a free cut of the original pack, a fair deal to eliminate all but one from their original free choice, a sealed prediction written at the start.

Now you reveal your prediction from the envelope…you predicted the card that is now face up on the table!

Magical mind reading…or is it?

To find out how to make the trick work and its link to Computer Science, go to the Magazine+ section of the webzine.

Killer Robot? Evil Scientist?! Helpless Woman?!?

(You can be the one to tell Angelina Jolie)

Lots of people think that Computer Science and IT are strictly for men only. That's really bizarre given that right from the start women like Grace Hopper and Ada Lovelace played pivotal roles in the development of computers, and women are still at the leading edge today. To be a successful modern IT Pro you have to be a good team player, not to mention great at dealing with clients, which are skills women are generally good at.

"Geeky male computer scientist" is of course just a stereotype, like "helpless female in need of rescue by male hunk", "scientist as mad eccentric in white coat", or "evil robot wanting to take over the world".

Where do false stereotypes come from? Films play a part in the way their (usually male, non-scientist) directors decide to represent characters.

Students on a "Gender in Computer Science" course at Siena College in the US watched lots of films with Computer Science plots from as far back as 1928 to

see how the way women, computers and computer scientists are portrayed has changed over time. Here are their views on some of those films.

1928 Metropolis

In a city of the future the ruling elite live in luxury while the workers live poorly in the underworld. An evil scientist substitutes a robot for a female worker activist. It purposely starts a riot as an excuse so reprisals can be taken. All hell breaks loose until the male hero comes to the rescue…

Computers	**X**	**Evil**
Women as IT Pros	**X**	**Helpless**
Computer Scientists	**X**	**Evil**

"Women are more or less portrayed as helpless … The computer scientist … as evil"

1956 Forbidden Planet

An all-male crew travel to Altair-4 to discover the fate of the colony there. They discover all that is left is scientist Dr Morbius, his beautiful daughter Altaira and a servent robot called Robby, programmed to be unable to harm humans. But what have Morbius' machines and experiments to do with the colony's fate?

Computers	**√**	**Helpful & Harmless**
Women as IT Pros	**X**	**love interest**
Computer Scientists	**X**	**Evil**

"Altaira plays a typical woman's role…helpless…unintelligent …Barbie-like"

1971 THX 1138

In an Orwellian future, an android controlled police state where everyone is made to take drugs that suppress emotion. LUH 3417 and THX 1138 stop taking their drugs, fall in love and try to escape…

Computers	**X**	**Evil Police**
Women as IT Pros	**X**	**Few**
Computer Scientists	**X**	**Heartless**

"The computer scientists are depicted as boring, heartless and easily confused"

1982 Blade Runner

In the industrial wastelands of a future Los Angeles, large companies have all the power. Robotic 'Replicants' are almost indistinguishable from humans but have incredible strength and no emotions. Deckard (Harrison Ford) must find and destroy a group of Replicants that have developed emotions and so threaten humanity as they rebel against being 'slaves'.

Computers	**X**	**Evil**
Women as IT Pros	**X**	**None**
Computer Scientists	**X**	**Caused the problem**

"A woman plays the minor role of a replicant…but is portrayed as a topless dancer"

1995 Hackers

A group of genius teenage hackers become the target of the FBI after they stumble across a high-tech embezzling scheme that is likely to cause a horrific environmental disaster. Dade Murphy and Kate Libby (Angelina Jolie) square off in a battle of the sexes and computer skills.

Computers	**X**	**Used illegally**
Women as IT Pros	**√**	**Elite…but illegal**
Computer Scientists	**X**	**Criminals**

"Angelina plays a hard hitting, elite hacker who is better than everyone in her group except Dane who is her equal"

The Real Pros

So much for fiction …what about real Women IT Pros? Equalitec (www.equalitec.org.uk) asked a few about their jobs. Here is what the real professionals think are the best things about what they do…

"Creating the initial ideas, forming the game, making the story… Being part of the creative process and having a hands on approach",
– Nana Louise Nielsen, Senior Game Designer, Sumo Digital

"Working with customers to solve their problems. The best feeling in the world is when you leave … knowing you've just made a huge difference."
– Hannah Parker, IT Consultant, IBM

"It changes so often… I am not always sure what the day will be like"
– Madleina Scheidegger, Software Engineer, Google.

"I enjoy being able to work from home"
– Megan Beynon, Software Engineer, IBM

"I love to see our plans come together with another service going live and the first positive user feedback coming in"
– Kerstin Kleese van Dam, Head of Data Management, CCLRC

" …a good experienced team around me focused on delivering results"
– Anita King, Senior Project Manager, Metropolitan Police Service

"I get to work with literally every single department in the organisation."
– Jemima Rellie, Head of Digital Programme, Tate

Remember stereotypes are fiction, careers are what you make of them and real robots are (usually) nice!

Even the dolphins use pocket switched networks

Email, texting, Instant Messaging, Instant response...one of the things about modern telecoms is that they fuel our desire to “talk” to people anytime, anywhere, instantly. The old kind of mail is dismissed as “snail mail”. A slow network is a frustrating network. So why would anyone be remotely interested in doing research into slow networks? Professor Jon Crowcroft of the University of Cambridge and his team are, and his kind of network could be the network of the future. The idea is already used by the dolphins - not so surprising I suppose given they are, according to Douglas Adams' “The HitchHiker's Guide to the Galaxy”, the second most intelligent species on Earth...after the mice.

From node to node

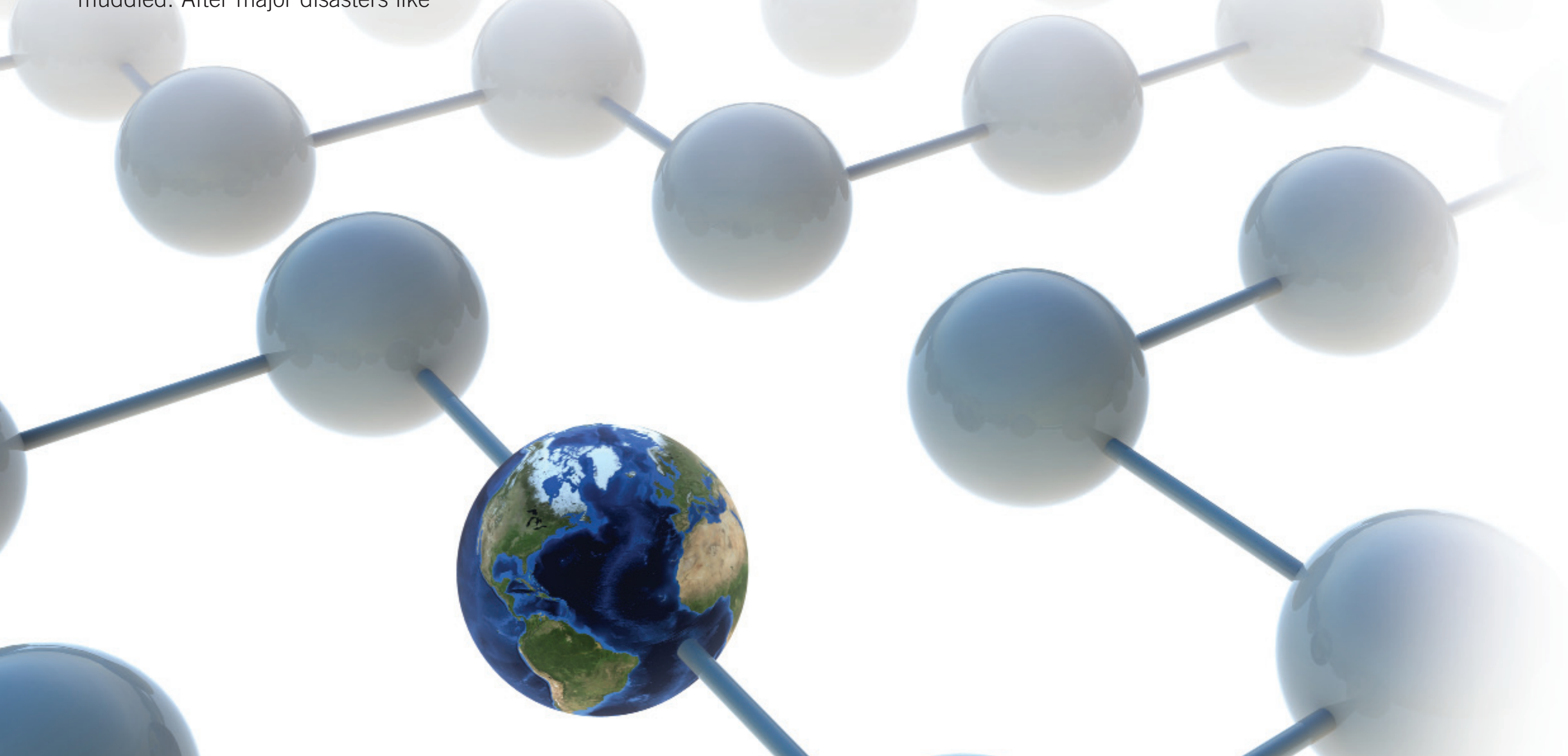
Traditional networks rely on having lots of fixed network ‘nodes’ with lots of fast links between them. These network nodes are just the computers that pass on the messages from one to the other until the messages reach their destinations. If one computer in the network fails, it doesn't matter too much because there are enough connections for the messages to be sent a different way.

There are some situations where it’s impractical to set up a network like this though: in outer space for example. The distances are so far that messages will take a long time – even light can only go so fast! Places like the Arctic Circle are another problem: vast areas with few people. Similarly, it's a problem under the sea. Signals don't carry very well through water so messages, if they arrive at all, can be muddled. After major disasters like Hurricane Katrina or a Tsunami there are also likely to be problems.

It is because of situations like these that computer scientists started talking about ‘DNTs’. The acronym can mean several similar things: Delay Tolerant Networks (like in space the network needs to cope with everything being slow), Disruption Tolerant Networks (e.g., in the deep sea where the links may come and go) or Disaster Tolerant Networks (like a Tsunami where lots of the network goes down at once). To design networks that work well in these situations you need to think in a different way. When you also take into account that computers have gone mobile – they no longer just sit on desks but are in our pockets or handbags, this leads to the idea of a ‘ferrying network’ or as Jon Crowcroft calls them: ‘Pocket Switched Networks’. The idea is to use the moving pocket computers to make up a completely new kind of network, where some of the time messages move around because the computers carrying them are moving themselves, not because the message itself is moving. As they move around they pass near other computers and can exchange messages, carrying a message on for someone else until it is near another computer it can jump to.

From Skidoo to you

How might such networks be useful in reality? Well one already exists for the reindeer farmers in the Arctic Circle. They roam vast icy wastelands on skidoos, following their reindeer. They are very isolated. There are no cell phone masts or Internet nodes and for long periods they do not meet other people at all. The area is also too large to set up a traditional network cheaply. How can they communicate with others?



They have set up a form of pocket switched network. Each carries a laptop on their skidoo. There is also a series of computers sitting in tarns spread around the icy landscape. When the reindeer farmers want a service, like delivering a message, the laptop stores the request until they pass within range of one of the other computers perhaps on someone else's skidoo. The computer then automatically passes the message on. The new laptop takes the message with it and might later pass a tarn, where the message hops again then waits till someone else passes by heading in the right direction. Eventually it makes a hop to a computer that passes within range of a network point connected to the Internet. It may take a while but the mail eventually gets through – and much faster than waiting for the farmer to be back in net contact directly.

Chatting with Dolphins

Even the dolphins are in on the act. US scientists wanted to monitor coastal water quality. They hit on the idea of strapping sensors onto dolphins that measure the quality wherever they go. Only problem is dolphins spend a lot of time in deep ocean where the results can't easily be sent back. The solution? Give them a normal (well dolphin adapted) cell phone. Their phone stores the results until it is in range of their service provider off the coast. By putting a receiver in the bays the dolphins return to most frequently, they can call home to pass on the data whenever there. The researchers encountered an unexpected problem though. The dolphin's memory cards kept inexplicably filling up. Eventually they realised this was because the dolphins kept taking trips across the Atlantic where they came in range of the European cell networks. The European telecom companies, being a friendly bunch, sent lots of text messages welcoming these newly appeared phones to their network. The memory cards were being clogged up with "Hellos"!

The Dolphin's memory cards were being clogged up with "Hellos"!

Jon's team are working on a pocket switched network for urban use (by the humans) called Haggle (See the webzine for more about it). So the dolphins may be the 'early adopters' of pocket switched networks but humans may be not far behind. If so it could completely change the way the telecom industry works...and if we (or the dolphins) ever do decide to head en-mass for the far reaches of the solar system, pocket switched networks like Haggle will really come into their own.

This article is based on a talk given at Queen Mary by Jon Crowcroft of Cambridge University.

Back (page) in Business

Designing for the disabled – that must be a niche market mustn't it? Actually no: the disabled have been the inspiration behind some of the biggest companies in the world.

Where do innovators get their ideas? Often they come from people driven to support those disadvantaged in society. The resulting technologies then not only help those with disabilities, but become the everyday objects we all rely on.

Alexander Graham Bell: Helping the deaf

Bell, one of the greatest inventors of the 19th century was inspired by his deaf mother. His life's work, that ultimately led to the invention of the telephone, was really about helping the deaf to learn to read and write. Bell's company went on to become one of America's greatest telecommunications giants. The sad irony of his work is that actually the telephone made things worse for the deaf. As telephones invaded the world of work, becoming a necessity to do business, many deaf employees were sacked because they could not use the phone.

Thomas Edison: Helping the blind

Thomas Edison is one of the most prolific innovators ever: so much so that his invention, the light bulb, is used to mean "innovation". Another of his inventions was the phonograph or record player, used universally before being overtaken by the digital age of CDs then MP3 players as the way to listen to music. In his patent for the phonograph, Edison put "listening to music" as only the fourth most important use. He was far more interested in reason 2: "the spoken book" to allow the blind to listen to books rather than have to read them. For many, listening to spoken books like the Harry Potter series while driving make the commute to work bearable.

Help the Disabled ...
Help us all

Temple Grandin and her squeeze-box

Temple Grandin is an animal scientist. As a result of her inventions and approach to animal welfare she has dramatically reduced animal suffering. She is heavily in demand from the meat packing industry to fix their problems. She is also autistic, and her career solving problems for animals started out with her designing a "squeeze-box" to help herself cope with autism.

Herman Hollerith and learning difficulties

People with disabilities have also been instrumental in the computer revolution. Early computers were controlled by punch cards – cards with holes in them that represent the data. The system was originally devised as a way of counting census data. Herman Hollerith founded a company that made his fortune around them. He had learning difficulties himself – a kind of cognitive disability. His personal motivation for developing punch card systems was so that people would not have to do the counting themselves. His company later became IBM, a company that helped propel us into the computer age.

So if you care about society or just want a source of innovative ideas, you could do worse than thinking of how to help take the disadvantage out of being disabled.

You may just make life better for everyone.

This article was inspired by a talk given at Dundee University by IBM's Vicki Hanson, a leading expert on designing for the disabled.

cs4fn is written and edited by Paul Curzon and Peter McOwan of Queen Mary, University of London. This issue has been supported by Equalitec (www.equalitec.org.uk) with additional support from ARM, Intel and Microsoft.

For more on the articles in this issue go to the Magazine+ section of the webzine www.cs4fn.org

Images kindly supplied by Futurelab, Yvonne Rogers, Kirstin Cater and Lewis Dartnell for this issue.

equalitec ADVANCING WOMEN

EPSRC Engineering and Physical Sciences Research Council

BCS

cs4fn is supported by EPSRC and the BCS

Microsoft ARM intel

Microsoft, ARM and Intel provide support for printing.

Queen Mary University of London