MACHAZINE

Volume 22 - Issue 4
July 2018

Wiskunde Informatica Studievereniging 'Christiaan Huygens'

JEROEN CASTELEIN

EvoSQL

HACKDELFT 2018

Go visit a hackathon

CHEERS

Dies 2018

FOKKO VAN DE BULT

Zeilburger's algorithm

CONTAINING:

CURRENT AFFAIRS | ASSOCIATION | COMPUTER SCIENCE | MATHEMATICS | MISCELLANEOUS

MACHAZINE
is a publication of
W.I.S.V. 'Christiaan Huygens'

Chief Editor
Dieuwer Hondelink, Rebecca Glans
Editorial staff
Marc Corstanje, Daniël van Gelder, Wouter Versteegh, Eva Slingerland
Art Directors
Jamy Mahabier, Beryl Sicam, Marjolein Leegwater
Contact address
Mekelweg 4, 2628 CD Delft
E: machazine@ch.tudelft.nl
T: 015-2782532
Concept and design
G_2O Kesteren
Publisher
GildePrint drukkerijen
Cover

Cooperating on this issue:

Advertiser index

Daniël van Gelder

You are reading the fourth issue of the MaCHazine of this year. As we are heading towards the end of the year and everyone is finishing up projects or preparing for exams, we can look back on a very busy year for everyone with lots of exciting activities and events. For me, this will be (for now) the last MaCHazine issue that I will be contributing to, since I will be doing my minor abroad in Singapore. Thankfully, the committee remains in good hands while I'm away.

Each issue we get a very wide range of articles which are provided by students, companies and staff at the TU Delft. It is always very interesting to read what all those people are up to and even more great to see their enthusiasm to take the time to write an extensive article about it in order to share it with us. In many cases this concerns research or large projects. It begs us to think about how large the fields of computer science and mathematics are. Often without people realizing it, these fields have a major impact on our daily lives and even more so as these fields keep growing by contributions of researchers.

With the growth of these fields, we see that the amount of layers of abstraction within technology get bigger as well. Just think about the amount of technology that is needed to transfer a single data packet a computer network. As a computer science student, I am constantly baffled by the amount of technologies and layers computers contain. We interact with a computer from a very abstract perspective. While using my computer, I see this document that I am typing, but the computer sees a stream of bits that it interprets and does all kinds of operations on. It is an interesting thought to consider how, even though humans invented everything that is in the computer, once you break it down or remove all the abstraction layers, nearly no one will have any clue what it is doing or how it works.

Hopefully, once I have completed my studies, I will be able to make a contribution to this field in some form or another, or at least understand some if its wide range of subjects. I hope that in some way this MaCHazine helps you understand more about it or at least introduces you to subjects that you don't know much about yet. Enjoy!

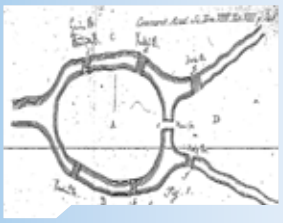
Table of Contents

Current Affairs

Association

Computer Science

Mathematics



Miscellaneous

From the Board

Marc Corstanje

Mathematics Education Affairs

Hi everyone! It's that time of the year again, the sun is starting to shine, weather is getting better and the final exams are closing in. For students this means that the holidays are near, but some stress is ahead of them. For me and my fellow board members however, it means something else; we are about to leave CH to our successors and take our hands of it. After a year of dedication, I can only guess that this is going to be a very hard task for us.

Recently, I was writing a piece for our yearbook. It had the theme `Bereik' which, mathematically translates to `range'. In Dutch however, it has another meaning as well. I was free to write about all we have `Bereikt', or achieved, this year. This turned out to be quite a lot that we, as a study association, should be quite proud of.

Since Mathematics Education Affairs was my function within the Board, my primary focus was typically on Mathematics Education. Within that field, we have focussed on the evaluation of the Master, the Bachelor, the minors and even the brand new Excellence Programme. As with every year, many things have changed again. We are changing into an English Bachelor next year, Algebra 1 is moving to Delft and Mathematical Modelling and Society in the Master is likely to undergo big changes next year.

Now unlike in the yearbook, I don't think this is a good moment to shine a light on all of the committees I have been in this year, but there is of course one I can't miss right now antdthat is the MaCHazine committee. We are in the fourth and last issue of this academic year and we have delivered amazing work throughout the year and I'm very proud of it.

For now, enough about the past. Let's look forward. A few weeks ago, our candidate Board 62, or EPA 62, announced itself to the world. Since they are hoping to become our successors next year, we are now making sure they are ready for it when the next academic year starts. This is already a lot of fun as we can regularly recognise ourselves in them. I enjoy watching them work so far and they seem to be having a lot of fun so far as well.

EPA 62 is not the only thing to look forward to in the coming time. Many students have been asking me what a Board member is going to do in the vacation, quite a good question actually as everyone is used to seeing me at CH all the time and CH is closed after the 6th of July. In July we will enjoy a good wel-deserved vacation. I'll be heading for London on the first free saturday where I'll visit Eric Clapton. After that I'll enjoy the sights in England and Portugal before going back home. In August it will be busy agian! The WIEWIE visits us regularly in August to work on the preparations for the Freshmen Weekend and the Study Visit Committee, one of my committees, will also join us to have everything prepared for the first General Assembly of Board 62 in which they will reveal the destination. To wrap it all up, in August we'll be at CH again and we'll be accompanied by manyu committees such as the WIEWIE, Wocky!, FaCie. FilmCrew, Study Visit Committee and of cours EPA will be there too.

Time to put an end to this story and get back to work. Good luck during the final exams and if you want to talk to a former Board member next year, I'll be sure to show up at the /Pub once in a while!

Current Affairs

TU Delft News

Wouter Versteegh, Editorial staff MaCHazine

The Delft University of Technology is the biggest and oldest public technical university in The Netherlands, established by King Willem II on January 8th, 1842. But what is currently happening in and around the TU Delft? This article will list the most important events of the recent months.

German President Frank-Walter Steinmeier visits TU Delft

On Tuesday May 15th, the president of Germany, Frank-Walter Steinmeier, his wife and a small delegation visited the TU Delft and Yes!Delft Students. It was part of an official visit to the Netherlands that took place on 15 and 16 May. Guided by Rector Magnificus Prof. Tim van der Hagen, the delegation visited presentations on cybersecurity, artificial intelligence, autonomous driving technology, and visited the QuTech laboratory for quantum technology. In these presentations, TU Delft professors explained to the president why it is so difficult to create an autonomous vehicle that can safely and effectively drive in heavy traffic, and they gave an introduction to scientific challenges regarding quantum computing at the QuTech laboratory. Lastly, they visited the Yes!Delft Students center for startups, where the delegation got a tour through the center and were shown a few startups at Yes!Delft.

Delft sensors to keep track of performance in wheelchair sports

Most of us are familiar with the Olympic Games, either the Olympic Summer or Winter Games. Similarly, most of us are also familiar with the Paralympic Games, which are the Olympic Games for people with disabilities, both physical and mental. For people with physical disabilities, a lot of sports are played using a wheelchair. For example, if you can't use your legs but still want to play tennis, you can play wheelchair tennis. And like any other sports, these athletes are also really motivated to optimize their performances. This inspired Delft researcher Rienk van der Slikke to develop a technique that monitors wheelchair mobility performance using small sensors, for which he received his PhD at the TU Delft on May 25th. According to Rienk, customization is even more important in wheelchair sports than in regular sports, because the performance is dependent on the interaction between the athlete, the wheelchair and the sport. The wheelchair is like an extension of the body, and assists the athlete in his efforts to improve performance and prevent injuries. The sensor he created, the Wheelchair Mobility Performance Monitor (WMPM), measures mobility as simple as possible, and has already been used in several wheelchair sports, such as tennis, rugby and basketball.

Test developed that predicts the degradation of paintings

Titanium white is a type of paint that was widely used in 20th century paintings by painters like Picasso and Mondriaan. The 'bad' version of this paint can lead to early aging of paintings when exposed to ultraviolet (UV) light. In collaboration with AkzoNobel, the Rijksmuseum and the Cultural Heritage Agency, TU Delft researcher Birgit van Driel developed a straightforward method to estimate the potential risk to the painting. She conducted research on the photocatalytic degradation of oil paintings containing titanium white. This degradation means that the work loses its intensity, color and luster, caused by UV light.

Free speech?!

Felienne Hermans, Software Engineering Research Group

Free speech. It is a topic I never really thought about until recent years. When I was young, free speech seemed like a good thing, a thing of the west, contrary to countries behind 'the iron curtain' or long gone times where the government controlled what we could say in public.

Now, when I hear people argue for 'free speech', like our own Dap Hartmann in last month's Delta, I cringe. Why? Isn't free speech good and important?

First, what happened?

A few weeks ago, a certain professor called a student with a beard *'a future IS terrorist that came here to practice'*. Footage of this incident ended up on Twitter, and then some people argued that this wasn't appropriate language to use in a classroom. TU agreed publicly and the professor was punished.

The end(?).

No, of course not. There are always people that think it is a good idea to rub more salt in the wound. In comes Dap Hartmann, brave defender of the free word, quick to note in Delta[1] that *"rabid feminists are at work here, who want to silence people who think differently, and hang them on the highest tree"*. He furthermore argues (as he has before) that there should be unconditional free speech everywhere, and that he wants a *"list of allowed jokes"*.

It ain't what you do (it's the way how you do it)

If people want something, in general it is good to take into account the context. If my seven year old fosterdaughter wants peanut butter, I am interested to know whether she plans to put it on a sandwich, or smear it on the floor. If people argue for more 'free speech', we should wonder on what or whom they are going to exercise it. This Terry Pratchett quote, for me, extends to all forms of expression:

"Satire is meant to ridicule power. If you are laughing at people who are hurt, it's not satire, it's bullying."

Free speech is meant to stand up against the government. Not to bully people. This sort of nuance is confusing to Dap and to many other 'betas'. The culture of a university of technology is so much based on rules, to calculate whether the spin is up or down, or whether a bit is 0 or 1, that we have a hard time dealing with things that are, by definition, not black and white. Things like using your gut feeling and admitting that you were wrong is not a common way of dealing with problems in our fields. This attitude can extend to how we deal with other issues, if we do not carefully consider them.

And what even is 'free speech'?

Since we are on the topic, let's have a look at what free speech even means. Many people wielding the 'free speech' sword misrepresent it, intentionally or not. If we actually take a look at our constitution[2], the words 'vrijheid van meningsuiting' (freedom of expression of opinions, i.e. free speech) do not even occur in the text. What is usually meant is article 7.1, which reads:

> *"Niemand heeft voorafgaand verlof nodig om door de drukpers gedachten of gevoelens te openbaren, behoudens ieders verantwoordelijkheid volgens de wet."*

Meaning that no one needs permission (from the government of course) to publish thoughts. Other means than press are addressed in a similar way in article 7.3.

That is it. The only thing it says, is we do not need permission from the state to print (7.1) and otherwise express (7.3) our opinions before we express them. After the fact, you may be punished for expressing your thoughts in several ways. For example we have laws against slander, against insulting the king and against inciting violence. People can (and are) regularly fined and even sentenced to jail for saying certain things. You cannot say whatever you want, and be free of consequences. That is not what our constitution (and many other) states. If you are somehow punished for expressing your thoughts, tough luck. Speaking is an action, and actions may have consequences. If people are upset with what you have said and complain to your friends or your employer about your remarks, these people are simply exercising their rights to express without permission, but do not, I repeat do not violate your constitutional rights. Once more for the people in the back: free speech is a thing between you and the government.

And TU Delft is not the government

What I may or may not do at TU Delft, is not simply determined by the law. An employer may ask you to be on time, to wear certain clothes and to behave in a certain way. Freedom of body (article 11), f.e. my right to wear visible piercings in my face, may be overruled by an employer. Similarly the right of free speech may be (and is) overruled by our code of ethics.[3]

Why? Because the relationship between a student and a professor is not that of equal adults. It is my responsibility to make sure students feel welcome and safe in the classroom, so safe even that they feel they can tell me if I crossed a line. I once addressed a student in a way that he did not feel comfortable with. He told me, and of course it hurt me, and my first impulse was to dismiss his feelings and think of him as too touchy. But he had a point, and I thanked him for the feedback and I tried to be better in the future.

Dap asks whether this *'is the direction we want to go in'*. I for one am proud that I work in a place where students call out our racist and sexist speech, and the university takes note and action.

[1] https://www.delta.tudelft.nl/article/column-jesuisherman

[2] http://wetten.overheid.nl/BWBR0001840/2017-11-17

[3] https://www.tudelft.nl/over-tu-delft/strategie/strategiedocumenten-tu-delft/integriteitsbeleid/code-of-ethics

Be careful, everything you say can be used against you!

Fred Vermolen, Associate Professor in Numerical Analysis

Nowadays jokes should be selected very carefully. In particular when you are being recorded, there is a risk for repercussions.

February 2018, a TU-Delft professor tries to silence a large group of first year students architecture at the Delft University. The students keep on talking and talking. Some students enter the lecture room walk in front of the lecturer while the professor tries to give his lecture. This behaviour is very offensive. The lecturer tries to focus on the lecture on history of architecture, but he cannot. He becomes more and more frustrated and at a certain point he is really fed up with the situation. He tries to silence the large group of students by some jokes. The camera keeps on running and records what he says and what he does. He decides to change his strategy by addressing some students in person. He points at one of the male students who does not pay attention to the course material by saying "I fully understand why you are being diverted because you are sitting next to a lady". Another student who does not want to be silenced is addressed by "Hey you there, with the IS-styled beard. You could go straight to the IS-caliphate right now and have some training there."It was clear that the professor was being sarcastic because he got tired of his students who did not want to be silenced. The footage can be seen on www.geenstijl.nl (https://www.geenstijl.nl/5141556/tu-delft-het-onderspit/).

The second joke regarding IS may not have been the best joke. On the other hand, if you give a lecture in front of a couple of hundreds of students who keep on making noise, then I could imagine that you do get frustrated. Possibly a better option for the lecturer was to leave the lecture room and tell the students to study the material on their own. I would probably have chosen the latter option. All these remarks were recorded on 'Collegerama'. As a result of this incident, the lecturer got an official letter from the Dean of the faculty saying that he gets an 'official warning' and that he has to apologize in front of the same group of students because of his remarks. Further he is no longer allowed to give this course to the students (see Delta, TU-Delft, https://www.delta.tudelft.nl/article/column-jesuisherman). Unfortunately, this is not where it ended. Activist Lawrence Cheuk posted a tweet where he expressed his dissatisfaction with the fact that the lecturer was not punished enough. The popular Dutch web blog Geenstijl wrote a blog about this matter as well. The reactions were not positive for the Delft University. The TU Delft press release replied by posting that the TU-Delft has given an official warning to the professor.

This story perfectly illustrates the vulnerability of lecturers. During the lectures people might make recordings of your lectures without you knowing it. Everybody has a smartphone or any other digital camera and these images could be shared on the internet. Further, recordings made by 'Collegerama' can apparently be shared (by making recordings of the recordings) on the world wide web. By the way, I wonder whether it is allowed to share recordings made by 'Collegerama' outside the Delft University. In this sense, all this worries me a little, because I also like to make some jokes during my lectures, and I know that my jokes could be on the edge sometimes. I hate being politically correct. Jokes that are on the edge are normally the ones that are appreciated the most. Do not get me wrong, I hate racism, I respect homosexuals, lesbians and I hate all forms of injustice. But I could make jokes on ethnicity, on homosexuals or even on religious people (despite being a Christian myself). This is the reason why I replied on Twitter to the TU Delft Twitter account by saying and wondering "This story demonstrates the vulnerability of lecturers. Is there an official document that tells us which jokes are appropriate and which jokes are inappropriate?" About 15 minutes after I posted this tweet, I received a phone call from the TU Delft press release. During this phone call I expressed my concern for my colleague.

What do I personally think of this story? Well, the joke about the student sitting next to the female was qualified as sexist by some people, whereas the joke on the IS-styled beard was deemed to be racist. It takes me an awful lot of imagination to see any racism or any sexism in these jokes. I do not see any correlation between men with beards and race, neither do I see any correlation between religion and race. I have seen Muslims, Christians, Jews and Buddhists of all races. Apparently the TU Delft feminists see this differently. As mentioned earlier, the joke about the IS-styled beard was not the best one, but I do not think that anyone should be punished officially for making bad jokes. Jokes in which you refer to your victim as an enemy of the state, such as an IS-terrorist or a nazi, are bad jokes and could be classified as not being respectful. However, the professor was not treated respectfully either and since he is a human being, I understand very well that he made this joke in his anger. In my opinion, the Dean has done a poor job by assigning an official warning to his professor and he should not have allowed this incident to be communicated to the outside world. In my opinion, the Dean should have backed up this professor and a private warning would have sufficed. In my perception, it appears that the Dean has lost his sense of reality in lecturing practice. He could have looked at the recordings of 'Collegerama' (he may have done so), and the students who did not want to silence during the lecture could have been warned as well. Further, the student who sent the recordings of 'Collegerama' to activists like Lawrence Cheuk could have been warned or even be punished as well. I even think that the student who shared the 'Collegerama' recordings possibly trespassed the law and in particular that he possibly violated the professor's portrait rights, that means it is not allowed to make a picture/film of someone and share it on the internet without his/her consent. In my opinion, it is wrong to punish the easy target (that is the lecturer) because his name is known. I would never have accepted to be treated like this.

Despite all this, I really enjoy lecturing! I am proud of working at DIAM and of our mathematics educational programs (BSc and MSc) and I am proud of our students. All this will not stop me from making (politically incorrect) jokes during lectures. If you feel uncomfortable with any joke I have made come to me and we will discuss the matter. So, dear students, do not worry, I will not stop. I think it is necessary to teach in a passionate way and this also requires some jokes because it is very hard to keep on listening to old jerks during the whole day. If you want to read more about this matter, I refer to Dap Hartmann's column entitled '#JeSuisHerman' in the TU-Delft Delta.

Time for beer öl, and other alcoholic beverages, skål!

Internship for an International student?

Romi Kharisnawan, International Student Computer Science

Hello again in an international student's column with me, Romi Kharisnawan.

In this column I would like to share my internship experience during my master. I am a 2nd year master student of Computer Science in the Data Science and Technology track. In my first year, I decided to find a summer internship to taste the working environment in The Netherlands. It's a bit tricky when it comes to the international student status, especially when you're from outside the EU. But, I must say hard work never fails. So, if you are really interested in experiencing a glimpse of working in The Netherlands, I suggest you to invest some time (around 3 months before) and effort to prepare yourself.

What to prepare?

This is the common question which comes across every internship opportunities seeker. I must say that mentality is one of first things to prepare. Mentality to find a job, mentality to get through rejection, mentality to work in the summer when most of your friends are on a beautiful vacation. These are several things that you should have beforehand. There are plenty of opportunities available every year, both on campus or outside campus. Keep yourself updated with all announcements on the elevators, Brightspace, or any other events, such as EEMCS day. As a short story, I got my internship in Philips Lighting from EEMCS day. But, you can also actively search opportunities on LinkedIn or other job portals. My tip is always to find your strength and interest, then find the available position and look at the requirements. Some internships require you to have EU citizenship or Dutch language proficiency. Prepare your best, especially for your dream company, because sometimes you can continue working on your thesis there or even work full time there after graduation. I got rejected once because I applied to a consultancy company and it required candidates to speak Dutch but I was so stubborn to apply for it. I always think that I might be lucky and get selected even though I don't possess one of the requirements. You can follow my path and maybe you'll have better fortune than me :D.

Currently, there are some companies offering "summer school" programs, such as Philips Lighting, IBM, Accenture, and so on. I think these kinds of programs allow people with all nationalities and only require English as communication language. The advantage of joining the program of a big enterprise is clear structure and curriculum. You will be given a real challenge and have the opportunity to present what you have done to the board of management at the end of your internship. Another option could be working at a startup where you will be working on their core product/service, where in enterprise you will usually solve small parts of a problem in the company. If you are more flexible in the working place, I think working in the UK or Ireland could be a nice choice, since English is their main language. Big tech companies, like Google or Facebook have a big office there. Beware that the selection for these companies will be challenging *(as we already knew :D)*.

During my internship, 15 people were chosen for the program and they were divided into 4 different groups. Each group had a mixture of backgrounds: computer science, business, design, and other technical disciplines. I was assigned to the Professional Space group where we created business value by improving safety without compromising on productivity in the factory. I found it really nice that in the program we have to know well about the context and motivation of the project in the beginning, which is not always applicable in university. The problem that I and my team tried to solve was avoiding collision between human and forklift or forklift and forklift. We applied an Indoor Positioning System (like GPS using lighting with higher accuracy) in this case. By getting to know the positions of each object/ person in the plant/ factory, we could provide warnings if they would bump into each other. I was in charge of the prototype development with another colleague from embedded systems. We built the prototype in their research lab using their luminaires by developing mobile apps, backend and the dashboard. It was a challenging yet fun task to do since we only had about 7 weeks to develop a prototype and it had to be working and presented to management. This was the first time that I used a full agile approach for a project and it turned out well.

As a final remark, I want to say that having an internship during your master is not only beneficial for your resume, but also beneficial for yourself, since you get to experience working in The Netherlands. There are plenty of opportunities for international students as well. Prepare yourself by finding information, updating your CV and writing your motivation letter, and you will be ready to apply to your desired job. Who knows, maybe your internship job can be your future job after graduation!


Rubber DAkCie


1 euro

Association

Dies: CHeers

Louise Zwep, Dies Commitee

"CHEERS" was the word you could not ignore for one whole week. Our beautiful association turned one year older, 61, and of course you have to say CHeers to that! That is why we, the Dies (birthday) committee, organized a whole week of fun activities. To get in the mood prior to the Dies-week, we organized a pre-pre-Dies, where we handed out cake and announced our theme: "CHeers". We also organized a pre-Dies, where you could take fabulous pictures in our Dies-photo booth. Then the Dies-week itself began, from March 26th to March 29th.

The kickoff was on Monday on which the chairmen of CH and the Dies committee gave a beautiful speech, along with an honorary member of CH. All this while enjoying more cake. In the afternoon there was a reception where you could congratulate the board with their 61th birthday. Lots of nice presents where given. After that, a big group of the association along with the Honorary Members went to have dinner. This was a lovely closing of the first day.

On Tuesday, we had a very special lunch-lecture. A professional CHEERleader came over to tell about her life and her profession. After hearing her interesting story, we also had to do some cheerleading ourselves. Within no time the whole group changed in a great cheerleading group.

In the afternoon there was even more reason to say "CHeers!". We went on a "bierfiets" which is a bar on a bicycle. On this we biked through Delft. It was a very fun ride with a lot of music, drinks and joy. We went with 3 "bierfietsen" in a row and crossed even the high-speed roads, even though the other cars did not like it that much...

Wednesday night was a very special night. For the first time in the Dies-history there was a craft beers night with live music! 5 craft beers were added to the /Pub collection. With the sound of live guitars and singers, everyone enjoyed the most delicious craft beers from all over the Netherlands! A very good reason to say "CHeers!".

After the nice drinks on Wednesday it was time for a very much needed hangover brunch in the /Pub. The Dies committee organized a special members lunch with sandwiches, hangover smoothies and champagne. This was a perfect combination to CHEER up and have a great day.

Unfortunately ,it was then time for the last activity of the Dies-week. On Thursday afternoon we went discorollerskating! This activity took place in the Lorre, a club in Delft. It was a very CHeerful activity where everyone could show their best dance moves on skates. Even the bartenders were dancing around on skates! With a lot of people, music and skates it was a great ending of a beautiful week. A big group did not have enough of it yet, and went back to Lorre that evening to dance and CHeer all night!

All in all it was a great week with a lot of different activities, one even more CHEERfull than the other. This will be a week that we, the Dies committee, will never forget. So for the last time, we say CHEERS to a beautiful week!

MatCH Made in Heaven

Bastiaan Bakker, MatCH

Hello, my name is Bastiaan Bakker and at the start of this year me and five others were asked to be a part of the committee called the Match. The other members of this Christiaan Huygens committee are Doris Aafjes, Joost Gobbels, Lufther Kronstadt, Tim Huisman, Floor Straver, and we are led by Irene Vooijs from the board. We started with dividing the member functions. Lufther became the chairman, Joost became treasurer and Tim Huisman became the secretary. After these three functions were divided, I became commissioner of Sports, Floor became commissioner of Games and Doris became commissioner of Promotion.

Our committee has to organize three to four events per year. We are already in the last semester so we have already organized three of them. The first one was the poker tournament, the second one was the old school gym activity, and the third one was a Just Dance tournament. Our next and last activity will be a sports and relax day at the lake in Delft.

Poker tournament

Almost every year the Match organizes a Poker tournament, and it is a great success every time. This year all the tickets were sold out and all the money earned from sale went to the tournament's prize pool. When people came in they got a glass of champagne from our lovely ladies Floor and Doris., Lufther gave an opening speech and after that the tournament was ready to begin. Me, Joost, Floor and Tim were the dealers at a poker table, Lufther was dealer at our blackjack table, and Doris was the lady that had to spin the wheel. We made a fortune wheel where you could gamble the chips you could win. You could double them, get them returned or lose them all. There was also the chance that you had to do a challenge to get the chips back. Eventually the tournament ended and we had a first, second and third place. They all received a money prize. After the winning ceremony, the activity was over and we all went to bed.

Old School Gym

At the old school gym activity we rented two school gyms for around 60 people. We did a lot of games you would do at your elementary or high school. We played games like dodgeball, basketball, soccer and many more. It was a great success and we improved the fitness of everyone that came to the activity. After all the physical effort they received a bottle of water and fruit to recover from all the hard work.

Just dance and day at the lake activity

In our recent Just Dance tournament there were a lot of pairs to compete for the prize and the title of best Just Dance pair of CH for this year. The tournament was in the /pub. At the day at the lake activity there will be a relaxed day with drinks, food and sports at the Delftse Hout.

Kind regards, the Match

Go visit a hackathon! *hacking is optional*

Paul Gerarts

This year W.I.S.V. 'Christiaan Huygens' organized its second hackathon, HackDelft. Students got a chance to work on datasets provided by our partners (Booking.com, MIcompany and Ortec Finance) and work with hardware sponsored by Major League Hacking.

After the opening Daniel Gebler gave a talk about what it was like starting Picnic and there was a Q&A with Ali Niknam from TransIP and bunq. After the introduction and a couple of workshops about the datasets everyone formed teams of about four people and started working on their projects. Some dove into mathematics, analyzing one of the datasets to assist marketing strategies, others created smartphone apps and user interface improvements for one of the partners, and some people even worked with IOT (internet of things) devices!

The teams had 24 hours, during which they could also take a look at some gadgets like VR headsets in the recreation area, play some games and take a bite to eat in the diner. Afterwards they could present their work to the judges and a few teams also presented their work on stage. The sponsoring partners talked a bit about what they liked about the projects and together with them we gave away some really cool prizes like IOT devices, microcomputers and online store credit. It was a really fun weekend and I would love to see it come back next year!

You may have noticed that so far I have not mentioned breaking into systems and software security. One of the things I noticed while we were organizing the hackathon was that a lot of people seem to think that a hackathon is about hacking into things. When you think about it, it is not all that surprising given that the term hacking is often used by the modern media as meaning 'using a computer to gain unauthorized access to data in a system'. However, hacking also means: "a piece of computer code providing a quick or inelegant solution to a particular problem" (thank you Oxford dictionary) which is what is meant with hacking at a hackathon.

At a hackathon you try to come up with a solution to a problem within a short amount of time, usually between 24 and 48 hours. Sometimes a problem is given to you and sometimes you are given data to come up with your own problem. These problems can be things like 'build an app that causes the users to have less of an impact on the environment' or 'make something that helps teachers teach'. Usually there is a lot of freedom and because of that hackathons are a lot of fun regardless of what skills you have. Can't write code? Maybe you can design a user interface, or create mathematical formulas. Maybe you are a musician and you can make a soundtrack for a game. Anyone can contribute something at a hackathon. You can go as a group or you can go alone and find some people to form a group with at the hackathon.

A hackathon is a great place to meet new people and learn new things. You will find people from other countries, speakers and representatives from companies, sometimes there will even be some hardware to tinker with, and there are usually some great prizes to win! I can recommend it to anyone who likes to make and develop to just go to one and see what happens.

KinderCHoCo

Willemijn Tutuarima

Every year, students leave their parents' house to go to university. This brings a lot of new things like meeting new people, getting to know the city and finding a place to live. Throughout the first year, a lot of these things get into the regular routine of the students, but some of the parents have no idea what their son or daughter does in Delft and where they spend their time. To get rid of this mystery, the CHoCo organized the parents' day. The committee, consisting of 12 freshmen students and 2 board members, worked from the beginning of the year to make sure this day was a big success (and a lot of fun). Next to parents' day, we also organized a dinner for the freshmen.

The first time we met with the committee we divided all the member functions. This way, everyone would have their own tasks. We needed a chairman, secretary, treasurer and commissioners for the morning and afternoon program, dinner, promotion and food and drinks.

One of the first things to consider was our theme for the year, we wanted something that had to do with the parents day and something to do with the name of the committee. Soon enough the Kinder Surprise Eggs came to mind and our theme was chosen: KinderCHoCo. As soon as we knew this was our theme, we bought a white sweater for everyone and printed our theme and member function on them.

Freshmen dinner

From then on we had a meeting every week because our first event was already approaching quickly. Next to the parents' day we also organized the freshmen dinner, which was a dinner for all freshmen with nice food and drinks. The dinner was planned for December and we immediately started arranging the location and what food would be served. This eventually became the Scouting Paulus as location and an Italian buffet as food, so now we only needed to think about the decorations. To take our theme to the next level, we made the dinner an Easter dinner because everyone eats chocolate eggs with Easter and it would be different from every other regular Christmas dinner.

When most of the important things were arranged, we started with the promotion because we wanted there to be as many freshmen as possible and did not know how many students to expect because this was the first time the event took place. Eventually we sold 80 tickets and knew the location would be perfect. The day before and during the day of the event we did the last shopping of drinks and decorations and brought everything to the location. The food was delicious and everyone had a good time. After cleaning up and bringing the last stuff back to EEMCS, we ended the night together in the Ruif.

Parents' day

Soon after the freshmen dinner, we started organizing our main event on the 7th of April: the parents' day. We had to think about the lectures that would be given, the lunch, a tour through Delft and many more things. Because some parents came from far away, we wanted to make sure that they would be able to see as much as possible of the faculty, the studies and Delft.

Early in the morning all the committee members arrived at EEMCS to begin preparations such as making coffee and preparing the check-in counter. Around 9 o'clock, the first students arrived with their parents who walked to lecture hall Ampere for an introduction to CH and EEMCS and an overview of the entire day. After that, the group was split into 2, depending on the study of the student. Firstly, the Computer Science students were given a lecture by Stefan Hugtenburg and the Mathematics students were given a tour through the faculty and went to the roof. After the first round, the groups switched and the Mathematics students were given a lecture from Joost de Groot. The groups joined again for lunch, where everyone got a delicious sandwich from Leo van Vliet. Together, we went on our way to the center of the city for the tour through Delft. Coincidently, the parents' day turned out to be the first official summer day of the year and it was incredible weather. This made the boat tour through the canals even better and a lot of people sat town somewhere to have a drink. We ended the day with a pub quiz in the /pub and pizza with the committee.

The CHoCo had a good year, with both events being a great success. I had a lot of fun working with the other 11 students and meeting a lot of new people this year during the events. I would recommend it to anyone because it is not a lot of work and really fun to organize events for other students, and of course be part of the best committee of CH.

An overview on Career Activites

Irene Vooijs, Chief Commissioner of Career Affairs

Startups and Entrepeneurships

Throughout the year, several activities related to entrepeneurships have been organised. We started the academic year with Startup while Studying where three Computer Science students explained how they combined studying with their own startup and how they founded their company. One of them even completely quit his studies and explained why he did this. The talks were inspiring and showed all different ways to found a startup. Later on, a workshop on entrepeneurship was organised where Ellen van Andel thaught us about the Business Model Canvas. This is a way to visualize all different stakeholders of a product and lets you think about the ways to monetize your product or service.

Academic Career

When you aim for an academic career, the first thing to do after finishing your Master, is starting a PhD. But what does that exactly mean? During the PhD Panel, several PhD students explained why they wanted to do a PhD and what it is like. This what not the only panel that was organized this year. Students could also go to the graduation panel, where the proces of the graduation project was explained as well as the choice between graduating at a company or at the university. The audience came prepared with many questions for the panel, which made the event into a great succes!

Workshops

In several interactive workshops, students had the change to learn and improve on all kinds of soft skills. Time management, Networking and Job interview skills were trained in sessions during the lunch or in the evening by different kind of organizations. In the last quarter, a lecture about stress and pressure to perform for students is given by a TU Delft psychologist. Not a real 'soft skill', but it is a topic to which many students can relate and is valuable to everyone.

Mathematical Career Orientation Event

For the first time, a career event especially for mathematicians was organized. They could meet up with 8 different companies to talk about the opportunities for mathematicians.

Association

Freshman Algorithm Programming Contest

Sterre Lutz

On Wednesday the 2nd of May, six fearless teams of freshmen stepped forth to demonstrate their skills in algorithm programming. In an epic showdown called the Freshmen Algorithm Programming Contest, the teams worked for three hours straight on solving several challenging problems, concerning anything from prime numbers to beer.

The problems could be solved in almost every major programming language, after which they could be uploaded to a special environment, where they were tested for a large range of inputs. For every correct submission, the teams received a balloon in a color corresponding to the question. For every incorrect submission, the teams received a time penalty of 20 minutes. In case of a tie in the amount of solved problems, the total amount of time it had taken the team to solve them plus all the time penalties decided who had won the contest.

The best of the competing teams had only solved 4 out of the 9 problems and more than one problem was left unsolved by all the teams! One of those was finding the amount of trailing zeroes behind n! for a very big range of n. This was also one of the problems where the input range greatly increased the difficulty of the question, since it made it impossible to simply calculate the faculty first and then ascertain the number of trailing zeroes. Another problem with that difficulty was checking for a given input whether it is a prime. For smaller numbers, you could simply check for every number between two and the input whether the input is divisible by it, but for larger numbers, this solution exceeded the 30 second time limit. The trick was to only check divisibility by the numbers two up to the square root of the input.

Together with a friend, I decided to sign up and we gave it our best shot. We didn't actually prepare much for the contest, and we struggled a bit, which might be an understatement actually. But hey, at least the snacks were good. Although we came in last, we like to think of it like this: we did beat every freshmen that didn't sign up!

All in all, it was a great experience and we will definitely consider signing up for other programming contests as well. It was also interesting to see how almost every programming problem had an underlying mathematical problem. Next time we will definitely prepare more and look for a third team member in the maths department. So if you like math, snacks and winning, consider yourself recruited by team hibster!

The FAPC is not the only event the CHipCie committee organizes! In September next academic year, they will also organize the Delft Algorithm Programming Contest, in which teams of mixed years can compete together. The best teams of the DAPC are invited to the the Benelux Algorithm Programming Contest, then the North-Western Europe Regional Contest, and all the way up to the World Finals! Are you up for the challenge? Keep an eye on the CHipCie website!

Blockchain for IOT

Sioux

Blockchain technology is becoming increasingly popular. It is all over the technology news bulletins and lots of companies are looking for ways to apply Blockchain technologies to their existing business or products. At Sioux we are always learning about new technologies, varying from machine learning to robotics and VR.

We also wanted to learn about Blockchain technology. Learning in this case means not only knowing how to apply this new technology but also when and why regarding high tech systems. This article describes a research project, where Sioux colleagues worked on applying this technology to a working IOT setup.

Goal of the project

The goal of the project is to show that applying Blockchain technology is both technically and practically feasible. With the latter we mean that we can meet product requirements like security. Sioux has used an existing Sioux IOT platform to work on. This platform was developed last year, runs on a Raspberry Pi and has sensors for Humidity, CO2 and Temperature.

Expected features

- Decentralized applications provide certain features such as immutability and independency which can make people trust these applications more than centralized applications.
- It is possible to avoid using cloud services, (Visa, MasterCard), which are considered trust worthy to save money in transactions fees.

Approach

At the moment there are a lot of different Blockchains available, each having its (dis)advantages and each based on different types of proof. Before actually starting implementation, the right Blockchain technology had to be selected. After investigation Sioux decided to go for the reliable Ethereum Blockchain, because of the following reasons:

- Ethereum is proven technology. It's widely used, well documented.
- Ethereum allows to easily configure local networks.
- Ethereum is one of the few Blockchains that is configurable for the proof of authority (POA) protocol. This means that mining is not required and the computational effort to process transactions and create blocks is limited.

Sioux IoT PoA network

PoA Ethereum networks can be configured according to the needs of the system. Therefore the result was the creation of a Ethereum PoA network (Figure 1) that has three Ethereum sealer nodes:

- Node 1 connects to decentralized applications running in the network and decrypts information.
- Node 2 is connected to both the Blockchain explorer and the Blockchain wallet (the Blockchain explorer can be connected to any node as can be the wallet).
- Node PI is running on the Raspberry Pi with the Temperature, Humidity and CO2 sensors. It also runs a decentralized application that sends a series of transactions to the Blockchain with encrypted information regarding the data collected by the sensors.

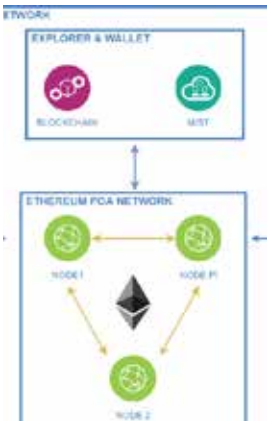


Figure 1

The decentralized application running on the Raspberry Pi is demonized (*"Deamon: computer program that runs as a background process"*), therefore it can never be stopped on a running device.

Once the information is stored in the Blockchain another decentralized application is created to:

- Decrypt the information being stored in the Blockchain.
- Store the real time data in a database. This data is accessible by the application that provides the graphical information coming from the sensors. (See Figure 2 & 3)

Figure 2

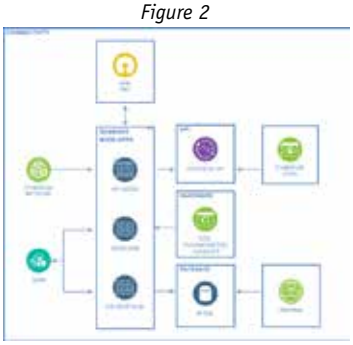


Figure 3

Next to this, we've implemented:

- A Blockchain explorer to keep track of all the information being stored in every block. (See Figure 4).
- A connection to a Ethereum wallet application for every node to keep track of the amount of Ether being spent and transferred as well as create new smart contract addresses if necessary. (See Figure 5)

Figure 4

Figure 5

Smart Contracts

In our setup – and in every Ethereum network – data is acquired, processed and stored. The nodes proof that the data is valid. How the data flows through the network and which entities are allowed to see and perform actions on this data, is described in smart contracts. These contracts describe the rules and terms for the usage of the data by whom; just like a real contract.

In our case, the smart contract allows to keep track of the status of the program that is collecting the information. It also ensures the data can only be changed by the owner of the smart contract – the PI node.

How to benefit from this

Smart Contracts allow decentralized applications to perform anything a centralized application does. Because these decentralized applications are able to use data from different sources in the Ethereum network, the possibilities go even further. This can be done using real Ether or using self-defined tokens in a local network.

This especially is beneficial when:

- Keeping track of program statuses is important.
- Unchangeable permissions are necessary.
- Secure transactions and secure encryption of any kind of information is needed.
- Independency is needed for applications to run free, irregardless of what is happening in the network.

Another advantage is that you don't need to rely on cloud services, which can be very costly. For example using Blockchain Technology to process payments and keep balances can be much cheaper than using existing technologies used by credit card companies, like Visa or Mastercard.

Conclusion

Blockchain Technology - and in our case a PoA Ethereum Blockchain - is technically ready to be implemented for real world situations. It offers features that are beneficial in a lot of existing situations. Whether you are looking to stop using cloud services for payments because the costs involved or you want to ensure independency amongst devices in client-server applications, Blockchain Technology can be the answer. We believe that the concept implemented in our IOT setup can be translated easily to any kind of application in the real world. It all depends on how smart contracts are designed and the purpose of the application for which they are deployed.

Graduation internship

We have managed to set up a PoA Ethereum network and have proven its feasibility. Should you be interested to continue this research for your Master thesis than don't hesitate to apply. We are open to hear any additional hypothesis you would investigate. We can imagine that interesting additional topics could to investigate the performance boundaries and resource usage of more complex systems.

Applyin

If you have any thoughts on how to continue as a graduation intern, please apply at jobs@sioux.eu and let us know.

Sioux. We bring high-tech to life

Sioux's strength lies in the unique combination of high quality competencies in software, mechanics, optics, physics, mechatronics, electronics, mathematics and IoT solutions. With over 600 engineers, Sioux supports or acts as the R & D department of leading high-tech companies. Sioux is keen to take responsibility: from creating ideas in the conceptual phase up to the delivery of serial production. Sioux wants to add value to its clients and build innovative solutions that can contribute to a society that is smarter, safer, healthier, more enjoyable and more sustainable.

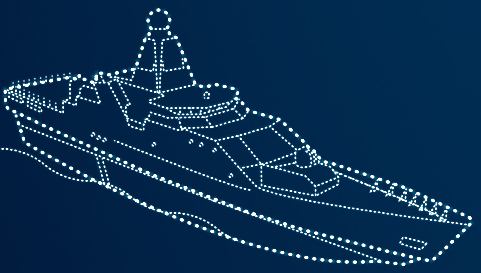
www.sioux.eu

JOIN US
IN EXPLORING A WORLD OF POSSIBILITIES

thalescareers.nl

LOCATED IN HENGELO, HUIZEN, EINDHOVEN, DELFT AND ENSCHEDE

100 INTERNSHIPS AND GRADUATION ASSIGNMENTS EVERY YEAR



THE MOST ATTRACTIVE EMPLOYER OF HIGH TECH JOBS IN THE FIELD OF SAFETY AND SECURITY

ACTIVE IN DEFENCE, TRANSPORTATION SYSTEMS AND CYBER SECURITY

THALES

Together • Safer • Everywhere

ComputerScience

BEP: Classification of diverse bacterial populations

Jasper Uljee and Yorick de Vries

During the second quarter of this academic year we worked on our BEP (Bachelor End Project) as part of our BSc Computer Science curriculum. We performed research for the Delft Bioinformatics department with Christine Anyansi and Thomas Abeel as our supervisors.

Introduction

When a patient suffers from an infection by multiple strains of a certain bacterial species, providing accurate treatment is a challenge. Bacteria playing a role herein can genetically be very similar but can have specific mutations causing differences in disease progression, antibiotic susceptibility or virulence. When one does not take such mixed infections into account while making a diagnosis, the outcome of treatment is uncertain and can be poor. There is a risk that a highly virulent strain (with possibly a relatively low abundance) remains untreated, while the patient is treated solely for another strain. Ideally, a mixed infection should be treated by antibiotics targeted at all pathogens present in the patient.

Classical techniques used for pathogenic bacterium identification rely on the isolation and enrichment of bacteria prior to analysis. Depending on the pathogen, this process can take up to several days before an accurate diagnosis is made and this is solely based on the isolated strain. However, when a patient is infected with multiple strains, faster and more accurate diagnosis by detecting individual strains is desired.

A suitable alternative to classical techniques to get insight in the microbial composition of samples is the use of metagenomic sequencing. Such genome sequencing techniques can give an overview of the complete metagenomic composition of a sample of interest and therefore provide a platform for mixed infection detection. Metagenomic sequencing provides the potential to obtain insightful knowledge about the microbial composition, as well as individual abundance levels. Treatment can therefore be better adapted to the different bacterial strains present.

Metagenomic tools

Currently several metagenomic analysis tools claim to accurately classify and quantify distinct bacterial strains within metagenomic sequencing datasets and it is of interest to what extent these existing genetic tools are able to do so. The tools that are currently used to detect and classify bacterial strains have not been independently tested and benchmarked. As the diagnosis of patients with mixed infections is impacted by the metagenomic analysis tools used, it is of importance to know the accuracy and performance of the tools used for strain-level metagenomic analysis. A common approach of these tools is to compare sequence reads against a reference database to detect and estimate the relative abundance of different strains with the usage of probabilistic models. A reference database with strains similar to what is expected in the sample is required.

Hereby we present an independent comprehensive benchmark of the most common tools for metagenomic analysis of bacterial samples at the strain level. The benchmarking strategy is applied with new, synthetic metagenomic readsets generated from available genome assemblies.

Material and methods

The performance of several specialized tools aiming at unraveling the microbial composition of metagenomic sequence data has been tested with a variety of inputs. To facilitate this, a framework was developed in Python 3 which enables the automatic creation and analysis of synthetic metagenomic sequence readsets. Synthetic readsets are desired as their properties can be precisely tweaked and the influence of these properties on the output of the tools can directly be measured.

Framework

In this framework (Figure 1), bulk readsets were generated from genome assemblies of the strains that should be included in the metagenomic samples. These reads are subsequently subsampled from the different portions of bulk reads to make metagenomic read pairs with desired properties. These batches of samples were varied in terms of species, amount of strains, distribution of the strains and coverage. Additionally, several experiments have been spiked with background noise in the form of reads from the Human microbiome project (HMP). Appropriate samples were made to investigate the influence of these parameters on the performance of the different metagenomics tools. Metagenomic tools used include: BIB, Pathoscope Sigma, StrainGR and Strainseeker. The tools BIB, Pathoscope and Sigma use complete genome reference databases to deduce the composition of the metagenomic samples. StrainGR and Strainseeker differ in their approach as these tools are matching the k-mer profile of the metagenomic reads to the k-mer profiles of the reference strains.

Bacterial strains

Metagenomic test samples were created as either a mix of Enterococcus (faecium and faecalis), Escherichia coli or Mycobacterium tuberculosis bacterial strains to represent different intra-species similarity. M. tuberculosis strains are known to be very similarto E. coli. Additionally two similar Enterococcus species have been chosen to represent samples with higher diversity between strains.

Evaluation

An appropriate measure to evaluate the tools' accuracy is how often the tools predict the correct strains (True positives, TP) and how often they make incorrect predictions (False positives, FP). Nonetheless it is just as important to measure the amount of strains that were in the sample but that were not predicted by the tool (False negatives, FN). These scores have been combined in the form of precision and recall, which can subsequently be combined to the F1 score:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1score = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$$

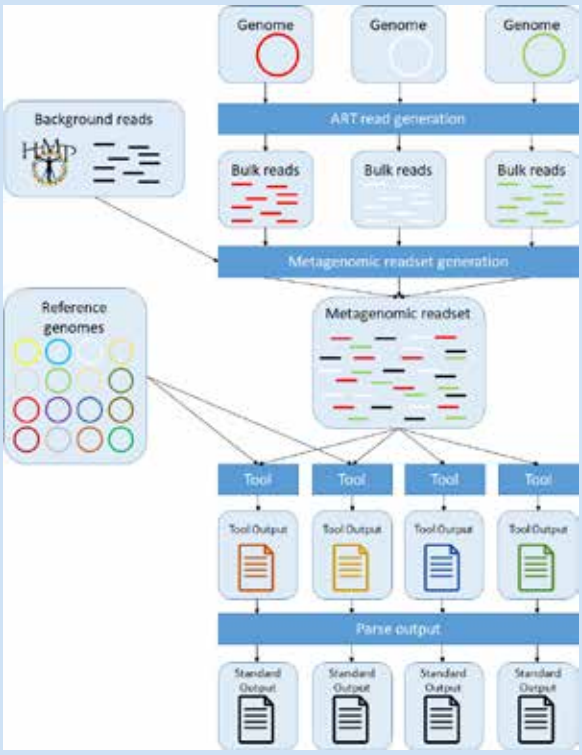


Figure 1: Workflow of the framework. This framework can automatically generate batches of metagenomic readsets with custom defined properties. The tools can easily do their analysis on those reads in a streamlined fashion. The outputs of the tools are put in a standardized format to make the comparison of tools easier.

Figure 2: Performance of the tools for different numbers of strains in the sample. The samples are all evenly distributed mixtures of strains where every strain is present at 1x coverage. Left: false positive and false negative scores (clipped at 20 strains). Right: F1 scores.

Results

Figure 2 shows the influence of the complexity of the metagenomics samples in terms of the amount of different strains present. It is expected that the complexity would negatively influence the performance of the tools because more strains need to be accurately distinguished. This is seen for StrainGR and StrainSeeker and to a limited extend for Sigma in the M. tuberculosis samples. However, the other tools are well able to identify more complex mixture of strains in the sample.

The BIB and Pathoscope tools have a quite consistent amount of false positive results in their output regardless of the amount of strains in the sample. For M. tuberculosis, BIB found less false positive strains in samples with higher complexity. This can also be related to the higher amount of reads present in samples with more strains. Sigma is well able to identify more complex mixtures of up to 8 strains, while having only a few false negatives when considering M. tuberculosis mixtures. In the case of StrainGR and Strainseeker, both tools are almost able to fully identify mixed strains in Enterococcus samples. For M. tuberculosis however, StrainGR is not able to predict multiple strains, while Strainseeker is able to do so.

Discussion

Mixtures of M. tuberculosis strains are harder to identify than mixtures of E. coli or Enterococcus. M. tuberculosis strains are more conserved among each other and are thus harder to distinguish in a mixed metagenomic sample. The tools were often similarly able to identify strains in mixed E. coli and Enterococcus metagenomics samples. For PathoScope and BIB however, the outputs for Enterococcus and M. tuberculosis readsets contained more false positives compared to E. coli. Most tools were well able to identify a high number of strains in more complex samples. Interestingly, while Pathoscope gives quite some false positive results, this is not related to the amount of strains present in the sample. This yields higher F1 scores for Pathoscope when analyzing more complex samples. StrainGR is able to correctly identify pure M. tuberculosis readsets, however, the tool is not able to correctly identify more strains in more complex samples, likely due to the high similarity of M. tuberculosis strains.

Conclusion

The current state of the art metagenomic tools are well able to give insight in the microbial composition of metagenomics readsets. With these tools, it is possible to get useful information about readsets, even in complex samples with up to 8 different strains present. However, mixed samples of the conserved species M. tuberculosis remain a challenge to fully unravel. Of the tested alignment based tools (Pathoscope, BIB and Sigma), Sigma consistently outperforms the other two while having a similar runtime. StrainSeeker and StrainGR, the two k-mer based tools investigated in this research, were found to perform similarly to each other. These tools additionally provide the advantage of a low runtime, desired for making quick diagnoses. However, these k-mer based tools do not provide as accurate results as Sigma. If you are interested in the full report of our project, you can read it here; https://repository.tudelft.nl/islandora/object/uuid%3A9580b976-fd95-4798-a13d-86ac479ab3eb

EvoSQL: Automated SQL Test Generation

Jeroen Castelein, Master Student Computer Science

The dream of any developer is that the applications they build are automatically tried and tested. Currently, having to build software tests by hand is one of the most important and time-consuming tasks. One of the paths to achieving this "dream world" is the use of automated test generation. For this purpose, tools such as EvoSuite[1] were introduced. EvoSuite takes Java source code and generates tests with the goal of maximizing code coverage. This way, each line of code has at least one test, and these tests will fail if a developer alters the code and its functionality. However, Java programs don't only contain Java code; database interactions are also commonplace, with the most common database communication language being SQL. In my thesis, I offered an approach for automatically generating tests for SQL queries, and an accompanying tool EvoSQL[2].

Branch coverage testing in programming languages such as Java is done by selecting a certain path or branch within the code that you wish to follow and selecting input parameters that make the program execution follow this branch. For simple code, such as a single-condition if statement, this can be done by using static code analysis and mathematical solvers to find the input parameters. When it comes to more complex requirements, a popular approach is to use search algorithms, where the dimensions of all parameters are explored.

Of course, SQL is not a programming language and, in its basic form, is only used to construct single queries for databases. These do not follow a similar structure as Java when it comes to finding branches. However, researchers have come up with a way to dissect a SQL query into a set of branches in terms of data constraints [1], which can be written as SQL queries. Such a branch is covered by some data if, when applying the constraints on this data, there is some output. Finally, a test is then constructed by executing the original query on the test data and comparing the output. If the query is changed during development and its functionality changes, the query's output will change and the test shall fail.

[1] *http://www.evosuite.org/*

[2] *https://github.com/SERG-Delft/evosql*

The Challenge at Hand

This left us with the problem tackled in my thesis: how to generate test data that successfully covers a branch (represented by a SQL query). Similar to with programming languages, data for simple queries can be found through static analysis and solvers (e.g. a single WHERE condition requiring a column to be of some constant value). However, as soon as queries get complicated, combining subqueries, joins and other SQL constructs, this technique has its limitations. Therefore, we chose to use a search algorithm. Search algorithms are transparent to the query complexity, as it only has to generate the data and measure how well this data qualifies as a solution. Static analysis can always be integrated to improve the algorithm.

There is a wide selection of search algorithms to choose from. A very basic one is brute force, simply trying every possible solution in the search space until a solution is found. As you may imagine, this is not our best option, as we have a huge search space and we are also able to use the query to identify how good a possible solution is. The algorithm of our choice is a Genetic Algorithm. Its benefits over other algorithms are that it is able to always find a global optimum, while still optimizing locally. It does so by being able to jump around in the search space.

A genetic algorithm holds a population of possible solutions, called individuals. This population evolves naturally like a real-life ecosystem. Each individual is tested for its fitness, and strong individuals are used to create new individuals, in a similar idea to procreation. Taking the features from two parents, a child individual can be created using any combination of their parents' features. This process is called crossover. Further, a child may mutate as well. Figure 1 visualizes this process. These mechanics combined with a large enough population may spread wide across the search space and explore many optima.

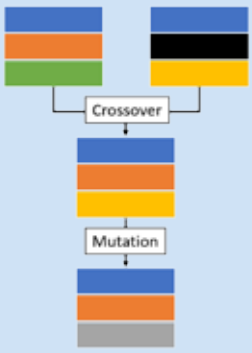


Figure 1. Visualization of the creation process of a new individual based on two parents.

EvoSQL's Genetic Algorithm

When implementing a Genetic Algorithm, it is important that its processes adapt well to the problem at hand. The first step is to generate the initial population. Here, it is important that this population is scattered throughout the search space. Each individual contains some data per table that is used in the SQL query. The standard option is to generate them at random. Further, by analyzing the SQL query we help the algorithm by creating some individuals that are likely solutions. This is done with care to avoid getting stuck in a local optimum.

The algorithm cycles until a solution is found, where in each cycle new children individuals are created to replace the population. As stated before, these are based on fit individuals from the current population. In EvoSQL this is a combination of their data tables. To ensure local optimization in the search space, an individual could be a full copy of one parent and then proceed to mutate. In this mutation a small change is made, with the aim of making this individual slightly better (or possibly slightly worse). Such a change could be as simple as incrementing some column value by 1. At the end of each cycle, the strongest half of the population survives.

The strength of genetic algorithms lies in the principle that after each cycle, the average fitness of the population is likely to be better and thus closer to finding a solution. This fully relies on a fitness function that correctly evaluates the entire search space, making it the greatest challenge to constructing a genetic algorithm. An optimal fitness function would give you a single funnel with a single optimum, in which a simple hill climbing algorithm will always find an optimal solution. With complex search spaces this is very hard to do, hence the use of algorithms broader than hill climbing. The fitness function we constructed for our problem uses the structure of the SQL query to calculate the fitness of an individual. This has multiple steps such as WHERE conditions, JOIN clauses, or subqueries.

Fitness Function

Implementing the fitness function thus requires us to run the query on the data in our individual, and then calculate the fitness based on how 'far' into the query the data reaches. To achieve this, we instrumented an existing open source database. By following the query execution through the database's execution flow, we extract which parts of the query are reached by the data. The example shows how this allows us to distinguish fitter individuals. We can then compare a population of individuals by stating that the farther that is reached, the fitter the individual. We fleshed out this process to deal with as many SQL constructs and datatypes we found necessary, whilst taking care to avoid local optima and oscillation.

Example
SELECT *
FROM orders
INNER JOIN products **ON** orders.product_id = products.id
WHERE orders.date = '2018-04-01'
As long as the join is not solved, the WHERE condition "orders.date = '2018-04-01'" is never evaluated. Thus, an individual in which the WHERE condition is evaluated is fitter than an individual where it is not evaluated.

Evaluation

To evaluate EvoSQL, a lot of queries had to be tested to ensure that our approach works well, and to allow for analysis of where our approach could still be improved. To ensure using a broad range of queries, we extracted over 2,000 SQL queries from open source as well as industry software. This collection of queries showed that there is a high presence of simple queries, however there are also enough complex queries to justify our approach. Another important part of the evaluation is that due to a Genetic Algorithm relying on randomness, our evaluation set must be executed ten times, with the results per query averaged over the ten runs. Finally, to compare our results, a baseline algorithm was also executed. This algorithm randomly generates data until a solution is found or too much time is consumed. This is a very simple search algorithm that we did not expect to be a challenge to our approach.
The results were positive; our implementation was highly effective in finding test data, achieving full coverage for 96.2% of the evaluated queries. The simple random search could do this for only 6.6%. Next, we performed a multivariate analysis to identify what variables may cause our approach to fail at finding solutions. From this analysis we saw that our approach struggled in solving string constraints, which can be accounted to the large multi-dimensional search space of a string. Finally, we discussed the performance and evaluated whether giving our approach more time could be beneficial to the coverage rate. Figure 2 shows the coverage achieved based on time given to algorithm, grouped by how many branches a query has, which indicates the difficulty of the query.

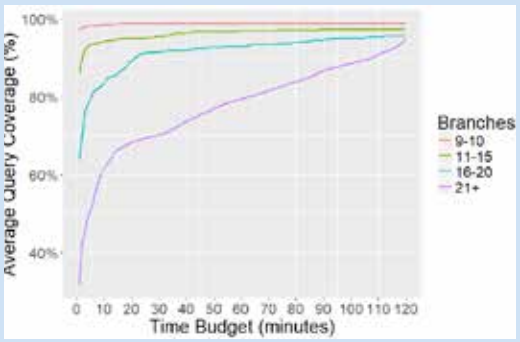


Figure 2

I had a great experience researching and writing the thesis. My supervisors were interested and invested in my project, which allowed us to discuss and improve a lot whenever new ideas or problems arose. With a successful evaluation we also decided to turn the thesis into a paper, which is published in ICSE '18 [2] and will be presented this May. This process included further evaluation and researching of which one of my supervisors, Maurício Aniche, took the lead. Here, a smarter algorithm to challenge our approach was also introduced to paint a more convincing picture that our approach is better than other plausible approaches. Finally, although the implementation of EvoSQL works well, it was not yet ready for developers to use easily in their projects. This is why two Honours Programme students have taken up the project of turning this into a faster, more complete, and more usable tool for developers. Hopefully, developers will be able to use this tool to improve their development cycle in the future.

[1] SQLFpc: Tuya, J., Suárez-Cabal, M. J., & De La Riva, C. (2010). Full predicate coverage for testing SQL database queries. Software Testing, Verification and Reliability, 20(3), 237-288.

[2] ICSE '18 Paper: Castelein, J., Aniche, M., Soltani, M., Panichella, A., van Deursen, A. (2018). Search-Based Test ata Generation for SQL queries.

Berth and shipping monitor for the Port of Amsterdam

Witteveen+Bos

In September 2016, we started a project for the Port of Amsterdam to investigate whether AIS data could be used to analyze berth[1] visits. The Port of Amsterdam wanted a tool which could be used to make investment decisions and gain insight in the port usage. The main purpose was to analyze what would happen if some berths with low usage were removed to lower the maintenance cost. On the other hand they wanted to check the occupancy rate for berths that are intended for a specific type of ship. This knowledge can be used to decide where new berths for this specific type of ship can be placed to lower the overall occupancy.

AIS data

The data that was used for the analysis is called AIS data. AIS (Automatic Identification System) is a worldwide implemented transponder technology used for safety reasons in the maritime sector. The transponder sends the locations of the ship as well as the ship characteristics such as length, width, name and MMSI-number (a unique identification tag). The signal has frequency between 2 and 6 seconds for moving ships and a frequency of 3 minutes for ships that are at anchor. These signals provide a large dataset which can be used for several analytical purposes.

Calculation definitions

The main challenge of this project was to transform the set of positions from the AIS data into an insight into port and berth usage. In multiple SCRUM sessions specialist (mathematicians, computer scientists and maritime experts) from both the Port of Amsterdam as well as Witteveen+Bos combined their knowledge to come up with a single definition. These definitions describes for example whether a ship is sailing, mooring or in berth based on the position and speed of the AIS data. Also the definition described what type of measurement errors are filtered from the input data and how the berth occupation is calculated, per area or per number of ships. You can imagine that if two small ships occupy the same space as a single larger ship the berth occupation is equal independent of number of ships.

Interactive dashboards

We created several dashboards using the business intelligence software Tableau. The development of these dashboards was also part of the SCRUM sessions. These dashboards give insight into the occupation of the berths, the movements within the port and the duration of a ships visit to the port. The user can also make comparisons between scenarios to gain insight into future developments of the port. All dashboards can be filtered by multiple input parameters such as ship type, ship length or visit period which give the user the possibility to zoom in on the data. Because the demands from the client are not always clear during a project, using Tableau we can easily adapt the dashboards, even in interactive sessions with the end users to fit their needs.

Figure 1: Dashboard

The dashboard shown in Figure 1 consists of 4 elements, filters to the left, an interactive map of the Port of Amsterdam at the top and two detailed graphs at the bottom. When a berth is selected in the map, the graph on the bottom left will display the occupancy rate for each day of the week (X) and each hour of the day (Y). In this example the asset is used most frequently from sunday until monday compared to fridays. The graph on the bottom right shows distribution that a berth is occupied with respect to time.

This project is an example of how using the right tool can save you a lot of time and frustration about continuously changing requirements. Part of using the SCRUM methodology is that the end goal is not always clear when starting a project. Separating the storage of data and the analysis from the visualizations gave us a lot of flexibility to quickly alter the dashboards without having to change the underlying data structure. The downside of this approach is the extra overhead of having to learn Tableau and adapting the data in Tableau for visualization.

[1] Berth a ship's allotted place at a wharf or dock

A58inbeeld.nl

In December 2018, we started developing a public consultation platform for the Innova58 project for Rijkswaterstaat. Rijkswaterstaat wanted to know what users and residents thought of the proposed widening of the A58 and wanted their opinion on a number of points of interests. In contrast to the project for the Port of Amsterdam our users weren't specialists wanting to gain insight but users of the A58 and residents living next to the A58. We consulted with GriDD, a company which is specialized in translating questions from engineers to an optimal user experience. They designed the user experience and created the content and we built the platform. The resulting platform can be viewed on https://a58inbeeld.nl.

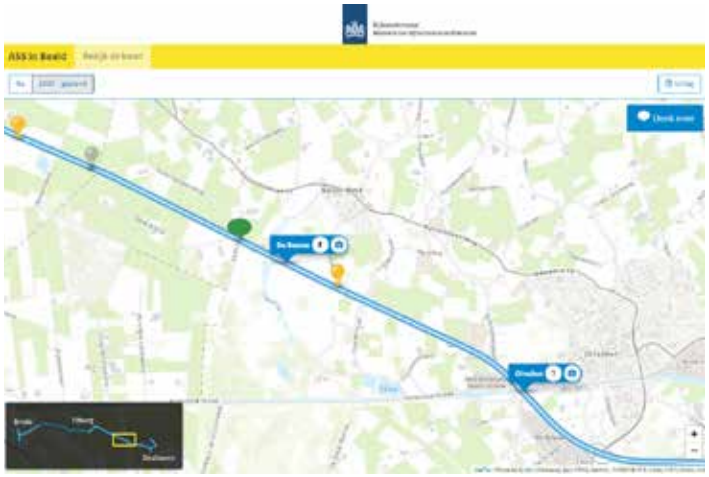


Figure 2 : Interactive map of the project area on which the user can place its comments

Architecture

The platform combines several types of data into a single web view: text and images on the home page and in the points of interest, geospatial data for the background layers and comments on the map and finally responses from Rijkswaterstaat. An option is to convert all data into a single database like PostgreSQL with a PostGIS extension. However, the problem with this approach is that we were responsible for the maintenance of the application and the database and every change would result in performing a new conversion to our database. Instead we let GIS specialists use their standard tool, ArcGIS and Rijkswaterstaat use their standard tool for project requirement management, Relatics. We then connected our server via api's to their data which meant that we do not have to write extra software to perform conversions and Rijkswaterstaat does not have to learn a new tool. This resulted in a reduction of the overall development costs and better processing of comments from users thus a better user experience. So being "lazy" as a developer can often result in a better experience for both the client and the user. The downside of this approach is that we had to work with an old and slow SOAP api but this was easily solved with a simple conversion script and server side caching.

Figure 3: Point of interest on which the user is asked for its opinion on a specific location

Further developments

The a58inbeeld website is still in development and we are currently working on converting the projects Envrionmental Impact Report (EIR) to a user friendly and interactive version. Usually the EIR was available as a large report with a formal procedure to comment on its contents. Our goal is to make the EIR more accessible by providing extra map layers with environmental impacts on which the user can place their comments. Another future improvement is using 360° Renders to show the proposed changes, users can place their comments in the 3d sphere which will give even more detailed feedback on a project.

This project is an example on how ICT can be used to improve public participations and create wider support for large projects such as innova58. Rijkswaterstaat has gained a lot of insight about the public opinion and also uses it during public consultation meetings.

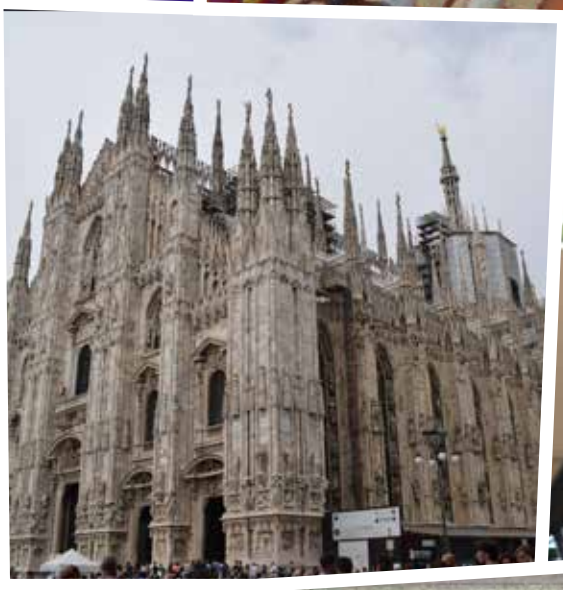
With these two projects we want to show you that our ICT projects for Witteveen+Bos range from big data analysis with a small number of specialist users in mind to web applications which support project decision making.

Léon Roeterdink and Mattijs Stam

Advertorial



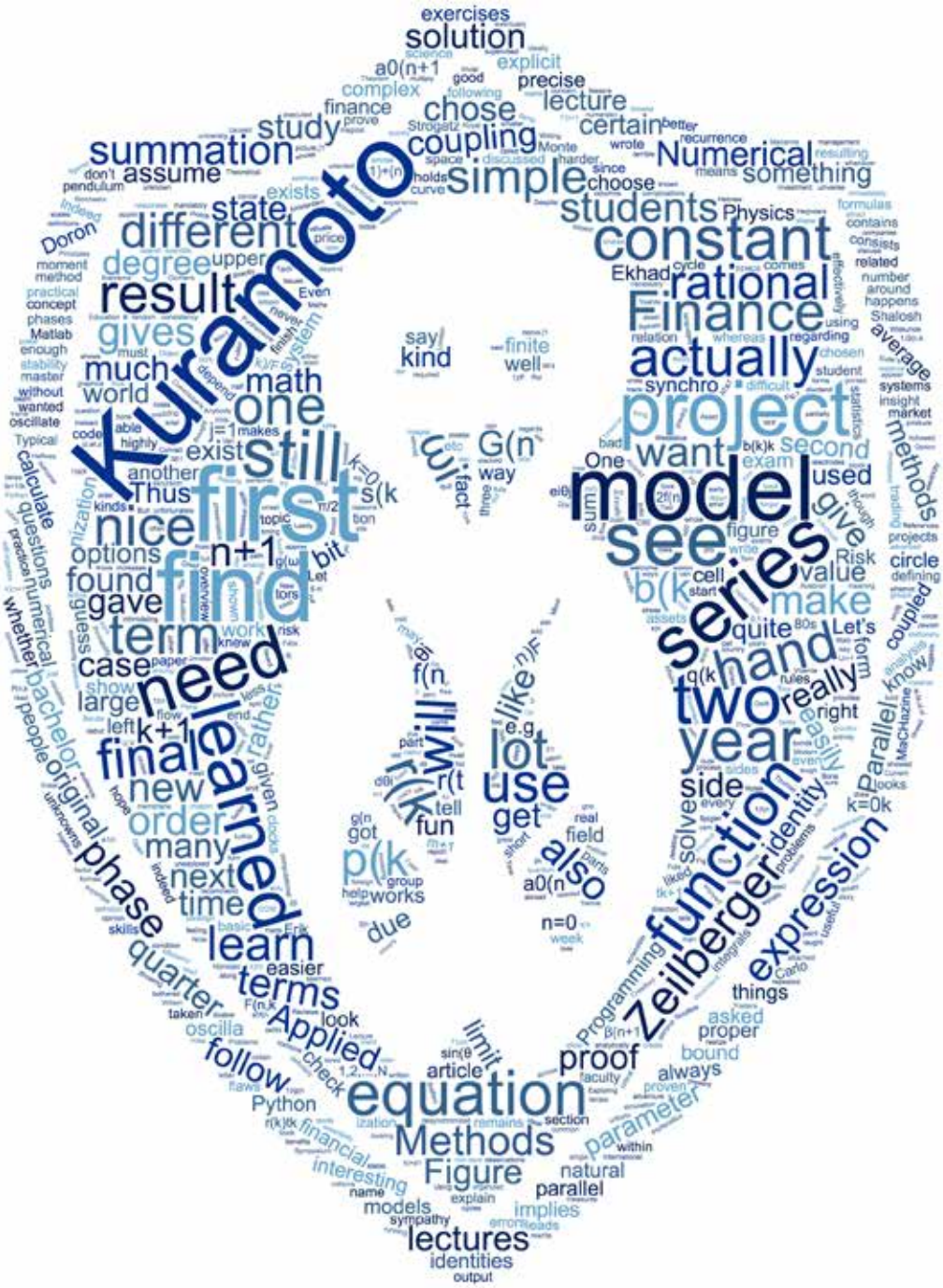




TU Delft
University Fund

Mathematics



Volume 22 • Issue 4 • June 2018

Synchronization: here, there and everywhere

Johan Dubbeldam, teacher (Mathematical Physics, Delft Institute of Applied Mathematics)

The word synchronization is commonly used to refer to processes whose frequencies self-organize. In this article I want to show some nice applications of synchronization and illustrate some of the mathematical techniques, especially the Kuramoto model, to investigate such systems.

Let's start with an introduction of the concept synchronization: *synchronization is an adjustment of the time scales of oscillations due to interaction between the oscillating processes.* The phenomenon of synchronization was originally discovered by *Christiaan Huygens* in 1665, a fact that cannot go unmentioned when writing for the MaCHazine, of course. In that year he wrote a letter to the Royal Society of London, where he coined the term "odd kind of sympathy" pertaining to the tendency of two pendulums to synchronize, or anti-synchronize, when mounted together on the same beam. A sketch of the system is shown in Fig.1.

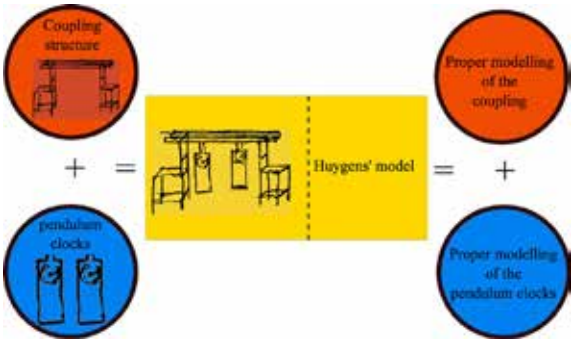


Figure 1: The original figure sketched by Christiaan Huygens in the middle; figure taken from [5]

Many more examples of synchronization were discovered later, such as synchronization in Josephson junctions, the flashing of fireflies, clapping of people, etc. The large variety of applications of synchronization suggests that there may be an underlying model that still captures the essentials but is rather simple. Such a model was put forward by Kuramoto. It is a rather drastic idealization of the synchronization phenomenon, but it gives a lot of insight in the phenomenon. In the next section I will introduce the model of Kuramoto, but first an essential key concept has to be explained: *self-sustained oscillations*.

Self-sustained oscillations

Before talking about synchronization, we need to be more precise about what kind of oscillations we are actually considering. The systems we consider exhibit what is called self-sustained oscillations. They have the following three properties:

- They do not damp
- They oscillate by themselves, that is, they don't require active forcing
- The shape and amplitude of the oscillations are chosen by the oscillating system and do not depend on the initial conditions

Typical examples of self-sustained oscillations, are grand-father pendulum clocks, the Van der Pol oscillator, heartbeats and oscillations in population models. Mathematically speaking, these oscillations correspond to an attracting *limit cycle* in the phase space, that is a closed curve to which trajectories in the vicinity are attracted.

The synchronization that we are interested in is that between self-sustained oscillations as described above. I will try to delineate the theory behind the coupled oscillations in the next section.

The Kuramoto model

As mentioned, the study of synchronization pertains to self-sustained oscillations and limit cycles in phase space. The most simple limit cycle one could think of is a circle. So Kuramoto [2] wrote the differential equation for movement on a circle for a large collection, say N, of oscillators as

$$\frac{d\theta_i}{dt} = \omega_i + K\sum_{j=1}^{N}\sin(\theta_j - \theta_i), \tag{1}$$

where $i = 1, 2, \ldots, N$ and K is a coupling constant between the oscillators; the ω_i are the natural frequencies of the oscillators. One can see that in the absence of coupling each oscillator will oscillate with its own natural frequency. The frequencies ω_i are distributed according to some distribution function $g(\omega)$ which Kuramoto considered to be unimodal and symmetric centered at $\omega = \bar{\omega}$; without loss of generality we can assume that $\bar{\omega} = 0$, in which case unimodality amounts to $g(\omega) = g(-\omega)$; see also the review paper [4].

Solving the model

The original analysis of synchronization was accomplished by Kuramoto [2]. The trick is to write Eq. (1) in a more convenient form by defining a first order parameter $r(t)$ as

$$r(t)e^{i\psi(t)} = \frac{1}{N}\sum_{j=1}^{N} e^{i\theta_j}. \tag{2}$$

Here $r(t) \in [0, 1]$ measures the coherence of the populations and $\psi(t)$ is the average phase. If all oscillators have the same phase θ, then $r(t) = 1$ and $\psi = \theta$. So $r = 1$ corresponds to a synchronous state, whereas $r = 0$ designates a non-synchronous state in which the phases θ_i are uncorrelated. Using the order parameter we can rewrite Eq. (1) as

$$\frac{d\theta_i}{dt} = \omega_i + Kr\sin(\psi - \theta_i), \quad i = 1, 2, \ldots, N. \tag{3}$$

This equation reflects that all oscillators are coupled to an average phase ψ. If we next go to a co-moving frame, we can actually make $\psi = 0$ and then we have a very simple ordinary differential equation which we know how to solve, if $r(t)$ is a constant.

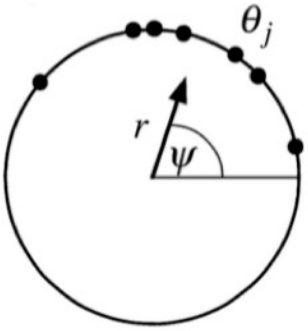


Figure 2: The average phase ψ and the oscillator phases θ_j on a circle

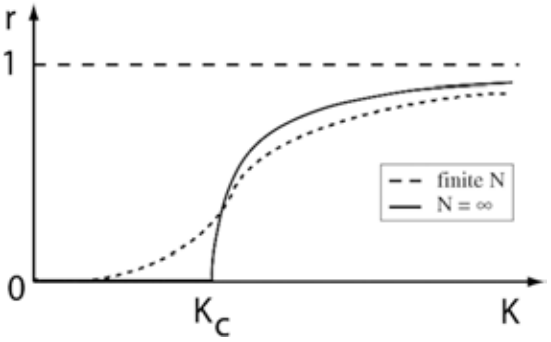


Figure 3: The order parameter r for different value of the coupling constant K. When $N \to \infty$ we find Kuramoto's result. For finite N the curve is more complicated

If we assume that the system actually reaches a stable state after some time, then we may indeed assume r constant. We can next find the equilibrium solutions of Eq. (3) with $\psi = 0$. There are actually two possibilities depending on the value of ω_i:

$$|\omega_i| = \begin{cases} \leq Kr & \text{synchronized} \\ > Kr & \text{not synchronized} \end{cases} \tag{4}$$

When all oscillators synchronize, hence $|\omega_i| \leq Kr$, for all $i = 1, 2, \ldots, N$, we have $r = 1$ and the state is fully synchronized. In the case that only a certain number of oscillators fulfills the synchronization condition, we have a partially synchronized state.

There is of course still one question left open: how can r be constant while the $\theta_i(t)$ are time dependent. Kuramoto conjectured that this was due to the fact that the oscillators have a stationary distribution.

To see that this really works, we calculate

$$r = \frac{1}{N}\sum_{j=1}^{N} e^{i\theta_j} = \int_{-Kr}^{Kr} \cos(\theta(\omega)) g(\omega) d\omega + \int_{|\omega|>Kr} \left[\int_0^{2\pi} \frac{C(\omega)}{|\omega - Kr\sin(\theta)|} \cos(\theta) d\theta\right] g(\omega) d\omega \tag{5}$$

The first contribution to r in Eq. (5) comes from the synchronized oscillators, the second contribution comes from the desynchronized oscillators. Even though the integrals in Eq. (5) look intimidating, we can easily do the calculation of the second integral and find its value to be zero. This leads to the following nice equation, required by the consistency of the mean-field approximation that we applied here.

$$r = rK \int_{-\pi/2}^{\pi/2} \cos^2(\theta) g\left(rK\sin(\theta)\right) d\theta \tag{6}$$

From this equation we can immediately see that we can find two solutions for r. A solution that is always there is $r = 0$. The other solution can be found when a specific function g is chosen.

We are, however, not so much interested in what the actually result looks like analytically, but a figure provides more insight.

In Fig. 3 you can see what happens with the order parameter as the coupling constant is increased. From a certain critical value of K, denoted as K_c, we find an additional solution of Eq.(1) besides the trivial $r = 0$ solution. This implies that for coupling strengths $K < K_c$, we do not observe synchronization, but for $K > K_c$ synchronization will be present. In order to get a better understanding of the model, numerical simulations can greatly help. You do not have to write the code yourself; there is a nice applet available on the internet [1]. An analysis of the stability of the states, which requires quite some work, can be found in the beautiful paper of Strogatz [3]. Before finishing I would like to draw your attention to some other problems.

Open problems

In this short article I have tried to give some impression of synchronization phenomena in general and the Kuramoto model in particular. A lot of questions related to this are still under investigation. Typical generalizations concern synchronization on complex networks. In this case the constant K can differ between two links, which makes the problem highly complex. Other synchronization research is concerned with so-called second order Kuramoto models. Such models are highly relevant for electricity networks where synchronization is important for the stability of the electric power grid. Yet another generalization of this model is to make a quantum analog of the Kuramoto model. The first steps have been taken into this direction, but this area is for the largest part unexplored. Anybody interested in complex networks and some kind of synchronization is always welcome to visit my office, so I can tell more about these projects.

References

[1] A nice simulation of the Kuramoto model can be found at http://rocs.hu-berlin.de/explorables/explorables/kuramoto

[2] Kuramoto, Yoshiki. H. Araki, ed. Lecture Notes in Physics, International Symposium on Mathematical Problems in Theoretical Physics. 39. Springer-Verlag, New York. p. 420. (1975)

[3] S.H. Strogatz. From Kuramoto to Crawford: Exploring the onset of synchronization in populations of coupled oscillators. Physica D 143, 1-20 (2000).

[4] Acebron, Juan A.; Bonilla, L. L.; Vicente, Pérez; Conrad, J.; Ritort, Félix; Spigler, Renato. "The Kuramoto model: A simple paradigm for synchronization phenomena". Reviews of Modern Physics. 77: 137–185, (2005)

[5] Pena R., Jonatan O, L. A., Nijmeijer, H. Alvarez, J., The sympathy of two pendulum clocks: beyond Huygens' observations, Scientific Reports 8 (2016)

Minor Finance

Tom Heijnders, Bachelor Student Applied Mathematics

When you choose your minor there are all kinds of crazy options. For example, you can go to a foreign country to take a minor abroad. Or you could follow courses at another university or faculty. These are all very exciting options if you're looking for an adventure. However, I chose something rather less exciting but still very interesting: I chose to follow the minor Finance at our own faculty EEMCS.

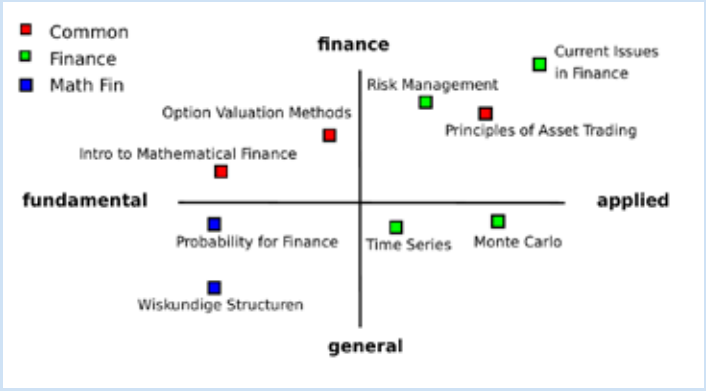
I have followed this minor in the 3rd year of my bachelor Applied Mathematics. I never knew what I wanted to do after I got my bachelor degree, so I wanted to use my minor to find a field where I could do my master in. When I came across the minor Finance, this really appealed to me. It seemed to be a fun way to use my math skills and apply it in the real world, and it was! Now I want to follow the finance track within our mathematics master. I'll give an overview about the courses you'll get in this minor and what you can expect along with them.

There are courses which are more applied to finance and don't need a lot of math. Here you'll learn about the ins and outs of the financial world. Other courses are more fundamental and you'll need a basic knowledge of statistics and probability.

Two courses I really liked were Intro to Mathematical Finance and Option Valuation Methods. During these courses you'll learn how to calculate the price of options and shares in different ways. You'll use some basic assumptions and quite simple math to valuate such a stock in the market. You'll do the same with options but then you'll eventually get some harder functions to work with.

Furthermore, there are 2 applied courses Monte Carlo Methods and Time Series. Here, you'll use Matlab and R to predict an asset path and produce random variables. You'll also use a lot of statistics and probability. But since all students have a different background in this topic, everything you'll need will be repeated. So it doesn't matter if you forgot most things.

There are two more applied courses where you don't need that much math: Principles of Asset trading and Risk Management . In this first course you'll learn among other things the meaning of assets and bonds and how to price them. In my opinion, this course was one of the most interesting courses. You'll use Excel to solve fun questions and use some functions to estimate



assets. During Risk Management you learn about the definition of risk and rules in the financial market to limit this risk. This was one of the harder courses as you had to learn a lot of rules and definitions.

Finally, we have the course Current Issues in Finance where once a week we discussed a new topic regarding the financial world. This year we discussed the Brexit, dividend taxes, bitcoin etc.

This last course is the only course without an exam or homework. For every other course there is a final exam and some courses also have mandatory homework. This hasn't bothered me because it's a good way to keep up with the course material. There aren't any group projects either. After all this, minor isn't very difficult. The minor is accessible for all programs within the TU Delft, so the math skills you need aren't that advanced.

Despite all the homework and exams, there was a fun activity too in which you could participate. This was a minor excursion organized by Bona Fide, a student investment association. They took us to Flow Traders and ING in Amsterdam. It was very exciting to see everything we learned in the real world. Both companies showed us around their trading floors and gave us a presentation about their expertise.

If the minor finance isn't enough of a challenge for you, you could also choose the minor Mathematics and Finance. The mathematics is a lot more difficult in this minor. It consists of almost the same courses as the minor Finance, but with two additional tough courses: Wiskundige Structuren and Probability for Finance. You can still follow this minor as an Applied Mathematics students, but you'll have to follow Monte Carlo and Risk management instead of Wiskundige Structuren. For a nice overview of the two minors see the diagram at the top of the page[1].

[1] https://www.tudelft.nl/en/eemcs/study/minors/finance/

Minor CSE

Erik Wesselius, Bachelor Student Applied Mathematics

Hi everyone! My name is Erik, I'm 22 years old and a fourth year Applied Mathematics bachelor student. This year was the debut of a new mathematics minor, Computational Science and Engineering (CSE).

To explain why I chose this minor, I have to tell you something about myself. I liked the course on programming in Python from the very first lecture. This gave me a feeling I could do something with what I learned. This is why I first chose the Computer Science minor, but that was not the thing for me. Instead of using some mathematics in computer programs, we learned how computer graphics work, a site works and whatnot. So after the first quarter I quit this minor.

At a certain moment that year, I heard from the COW (Commissaris Onderwijs Wiskunde or Applied Mathematics Education Affairs) that there would be a new mathematics minor, one which would combine programming and numerical methods. From that moment on I knew that this was the minor for me.

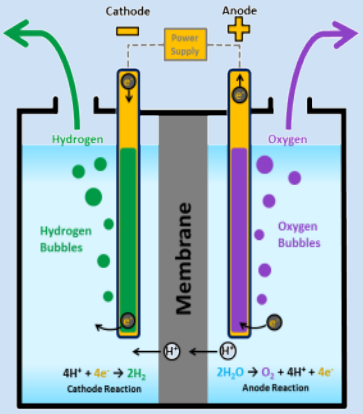
The minor itself contains 5 courses and 1 final project. The courses of the first quarter were Numerical Methods for Differential Equations, Scientific Programming, Parallel Computing and a literature study for the final project. The second quarter courses were Object oriented scientific programming with C++ and Numerical methods for Stochastic Differential Equations and the final project itself.

The Numerical Methods course was a version of the course Numerical Methods 2 from the Applied Mathematics bachelor. This course was a bit simplified for the students from different faculties, there were much less theorems to proof and more practical exercises to make. Even though this course is simplified, it isn't doable for students who have never had any course regarding numerical methods, e.g. Computer Science students.

The Scientific Programming course gave me even more fun then I'd thought. The course is split into 2 parts: lectures and practicals. In the lectures you learn most of the computer itself, how it makes different sort of errors, and during the practicals you actually see these errors occur and you're asked to fix them, given the knowledge from examples and the lecture notes. In this course we learned what the benefits are of C over Python or over Matlab and how we can program effectively in different kinds of programming languages. For instance, did you know you should iterate over the columns rather than the rows in Fortran and in C vice versa?

During the Parallel Computing course we learned that some programs could be a lot faster when some parts of the code are executed in parallel, e.g. certain loops. This was one of the many things we learned during this course. In this course we had practicals and lectures. In the lectures we were taught different methods of parallel computing, how we could implement them effectively, what you shouldn't do when programming in parallel and more. In the practicals we were given exercises about the lectures. So this course was quite practical, everything you learn in the lecture was directly applicable on the exercises.

Halfway the first quarter we made groups for the final project, I formed a group with Marianne Schaaphok (Mathematician), Matti Dreef (Electrical Engineering) and Thomas Oomens (Computer Science). With them we chose the project about alkaline electrolysis. This project was supervised by Willem Haverkort. He gave us a brief explanation about the project and a document with the necessary equations and explanations. In short, alkaline electrolysis is a way to create hydrogen using a solution, electrodes and a membrane. The process is described in the picture to the right of this paragraph[1].



The literature study had a bit of bad timing. During the first exam week, we needed to file both the report for Parallel Computing and we had about 1 or 2 weeks to start and finish the literature study. This gave me a lot of stress and I had to ask for a suspension of the due date. Because of this and personal reasons, I slacked a bit in the other courses and I had to drop the other 2 courses, so unfortunately I can't tell you about them.

However I did help finish the final project. In the project we were asked to model alkaline electrolysis and were asked to find out what happens if you add a flow in the cell. We used Python and everything we learned at Scientific Programming and Parallel Computing to program this and we used what we learned in Numerical Methods to approximate the cell itself. So we used everything what we learned in the first quarter in the project. In the end, we found that a flow increases the overall efficiency of the whole cell.

Lastly, this minor was a really nice experience and I learned a great deal about computational science and I recommend this to everyone that likes mathematics and computer science. This year there were some complications with, e.g. the final project and some minor flaws in courses. This is natural because it was the first year of the minor and I have faith these flaws will be fixed next year.

I hope you've enjoyed this article and if you have any questions about this minor, feel free to look me up.

[1] https://www.energy.gov/eere/fuelcells/hydrogen-production-electrolysis

Zeilberger's algorithm: Where computers meet human ingenuity

Fokko van de Bult, Lecturere and researcher in the Analysis group

Because the MaCHazine is for both Mathematics and Computer Science students, I figured it would be nice to discuss where computers and algorithms meet my own research. I study hypergeometric functions, which can most easily be described as series involving binomial coefficients and factorials, for example $\sum_{k=0}^{n} \binom{n}{k}$, or $\sum_{n=0}^{\infty} \frac{x^n}{n!}$. You want to know as much as possible about these series; what you ideally want is a formula such as $\sum_{k=0}^{n} \binom{n}{k} = 2^n$, where the series is explicitly given by a single term, or at least a finite number of terms independent of n. Sometimes this is impossible and we have to settle for identities as $\left(\sum_{n=0}^{\infty} \frac{(-x)^n}{n!}\right)\left(\sum_{n=0}^{\infty} \frac{x^n}{n!}\right) = 1$.

One result that I find amazing is that it is possible to determine whether or not a hypergeometric series has a summation in a simple form. This is related to integrals, where you can prove that the anti-derivative $\int e^{-x^2}\,dx$ does not have an expression in terms of elementary functions. Showing that something does not exist always seems a bit magical to me, but of course, in practice the fact that a computer can give you a summation whenever it does exist is more useful. Essentially the method consists of being able to give an upper bound to the complexity of the summation and having the computer check all simpler forms. If the computer does find a positive result, it leads to some amazingly deus-ex-machina proofs.
Of course, to be able to make a bold statement about whether many series can be evaluated or not, you have to be precise in what series you consider and what the outcome is allowed to look like. And before I lose half my audience by displaying complicated formulas, you have to realize that the difficulty of the formulas actually make it more useful. Indeed, the more parameters in the summand you allow, the more series you can possibly evaluate (or show a summation does not exist). Likewise, many parameters in the resulting expression gives a more impressive result.

Theorem 1. *We say that $g(n)$ is a hypergeometric term if $g(n+1)/g(n)$ is a rational function of n. This implies that*
We say $F(n,k)$ is a proper hypergeometric term if

$$F(n,k) = P(n,k)\frac{\prod_{i=1}^{U}\Gamma(a_i n + b_i k + c_i)}{\prod_{i=1}^{V}\Gamma(u_i n + v_i k + w_i)}x^k$$

with P a polynomial, $a_i, b_i, u_i, v_i \in \mathbb{Z}$, $c_i, w_i, x \in \mathbb{R}$, $U, V \in \mathbb{N}$, and Γ is Euler's gamma function ($\Gamma(n+1) = n!$). This implies that $F(n+1,k)/F(n,k)$ and $F(n,k+1)/F(n,k)$ are rational functions of n and k.
Let $F(n,k)$ be proper hypergeometric and suppose that for each n only finitely many terms in the series $f(n) = \sum_{k\in\mathbb{Z}} F(n,k)$ are non-zero, then there exists an explicit algorithm to determine whether an expression of $f(n)$ as a finite sum of hypergeometric terms exists, and if so, returning this expression.

The existence of such an algorithm is nice, but if it takes the age of the universe to calculate the summation of a relatively simple sum, it is still pointless. While the theoretical upper bound on the running time that can be proven tends to be very large, in practice current-day computers can easily evaluate the series of this kind that people are actually interested in. Moreover the output of the algorithm gives a proof of the identity which humans can check by hand, so we don't have to assume a result is true because "the computer said so".

Figure 1: Doron Zeilberger is definitely the man to call if you need to evaluate a complicated series.

One of the main components speeding up the algorithm was discovered by Doron Zeilberger, who is a quite vocal advocate for the idea that computers will soon be better at proving mathematics than humans. In a field which contains many smart people who got very far by being good at doing amazing calculations by hand, this has sometimes gotten some negative responses. For example, when the algorithm was first published, one of the distinguished professors in the field explained a beautiful proof of a new identity at a conference. At the end he remarked that it could be proven by Zeilberger's algorithm as well, and he had printed the proof on his (still physical) slides. Subsequently he let the 20 slides full of terrible calculations roll out over the lecture floor. It did turn out that a programming error had caused the output to be this bad. However, I guess the computer can show an identity holds, but not explain why it holds, which means that even with such an algorithm we need humans and computers.

Zeilberger's algorithm can not only prove known (or conjectured) identities, but also find some new ones. In the search for new identities, Zeilberger has often co-authored with Shalosh B. Ekhad. Ekhad is an amazing calculator who can quickly sift through millions of possibilities. Indeed, with 27 articles this must be one of the most published computers around. The name is Hebrew for 3B1, the early 80s desktop computer from AT&T, which was Zeilberger's original computer (note that Zeilberger is Jewish). He has his own webpage, which is interesting to browse through, on which you can find a self-written plane geometry textbook (with a foreword by Zeilberger), which is a computer-generated list of geometry-theorems with proofs.

Figure 2: Shalosh B. Ekhad is a frequent co-author of Doron Zeilberger. I hope he got some upgrades since this original 80s version.

Let's get back to Zeilberger's algorithm. To understand the method, let's consider $f(n) = \sum_{k=0}^{n} \binom{n}{k}$. As $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$ we find that

$$f(n) = \sum_{k=0}^{n} \binom{n}{k} = \sum_{k=0}^{n} \binom{n-1}{k-1} + \sum_{k=0}^{n} \binom{n-1}{k} = 2f(n-1),$$

where we shift the summation index k to $k+1$ in the first of the two series on the right hand side to see it equals $f(n-1)$ as well. From this recurrence and $f(0) = 1$, we can easily deduce the general formula $f(n) = 2^n$. The computer-generated proof of this identity is essentially the same. Rewriting the expression $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$ to $\binom{n}{k} - 2\binom{n-1}{k} = \binom{n-1}{k-1} - \binom{n-1}{k}$ we see that if we now sum both sides of the equation over all k (in $\mathbb{Z}$, where we use that $\binom{n}{k} = 0$ if $k < 0$ or $k > n$), the left hand side becomes $f(n) - 2f(n-1)$, and the right hand side telescopes to 0:

$$\sum_k \binom{n-1}{k-1} - \binom{n-1}{k} = \cdots + (\binom{n-1}{0} - \binom{n-1}{1}) + (\binom{n-1}{1} - \binom{n-1}{2}) + \cdots = 0,$$

as we can cancel the $\binom{n-1}{1}$ term, and all other terms as well. Thus we have found $f(n) - 2f(n-1) = 0$ as before. It can be shown that for every proper hypergeometric term $F(n,k)$ there exists a recurrence relation of the form

$$a_0(n)F(n,k) + a_1(n)F(n+1,k) + \cdots + a_J(n)F(n+J,k) = G(n,k+1) - G(n,k),$$

where the coefficients $a_i(n)$ are polynomials in n, and where $G(n,k)/F(n,k)$ is some rational function in n and k. In this case we have $J = 1$, $a_0 = -2$, $a_1 = 1$, $G(n,k) = -\binom{n}{k-1}$.
While there is an explicit upper bound for J which you can use, the smaller the J, the simpler the relation, the easier everything becomes. For our choice $F(n,k) = \binom{n}{k}$ we choose $J = 1$; easier still would be $J = 0$, but unfortunately that won't give a positive result. Let's denote the left hand side as t_k and thus we get

$$t_k = a_0\binom{n}{k} + a_1\binom{n+1}{k} = G(n,k+1) - G(n,k).$$

Here we suppress the n-dependence of t_k, a_0, and a_1. At this point we make the "guess" that $G(n,k)$ looks like t_k, to be precise $G(n,k) = r(k)t_k$, for some rational function $r(k)$ (of k and n). This is not really a guess as the theorem tells us that for large enough J such a rational function must exist. Then we have

$$t_k = r(k+1)t_{k+1} - r(k)t_k \quad \Rightarrow \quad 1 = r(k+1)\frac{t_{k+1}}{t_k} - r(k).$$

Let us consider t_{k+1}/t_k, this gives (multiply numerator and denominator by the common factor $(k+1)!(n+1-k)!/n!$)

$$\frac{t_{k+1}}{t_k} = \frac{a_0\binom{n}{k+1} + a_1\binom{n+1}{k+1}}{a_0\binom{n}{k} + a_1\binom{n+1}{k}} = \frac{a_0(n-k)(n+1-k) + a_1(n+1)(n+1-k)}{a_0(k+1)(n+1-k) + a_1(n+1)(k+1)}$$
$$= \frac{a_0(n-k) + a_1(n+1)}{a_0(n+1-k) + a_1(n+1)} \frac{n+1-k}{k+1} = \frac{p(k+1)}{p(k)} \frac{q(k)}{s(k)},$$

where

$$p(k) = a_0(n+1-k) + a_1(n+1), \qquad q(k) = n+1-k, \qquad s(k) = k+1.$$

Note that the unknown a_j's only appear in p. Plugging this into the equation $1 = r(k+1)t_{k+1}/t_k - r(k)$ gives

$$1 = r(k+1)\frac{p(k+1)}{p(k)}\frac{q(k)}{s(k)} - r(k),$$

which multiplied by $p(k)$ becomes

$$p(k) = r(k+1)p(k+1)\frac{q(k)}{s(k)} - r(k)p(k).$$

Remember that we want to solve for $r(k)$ (and need a_j's defining p allowing us to do so). Writing $b(k) = r(k)p(k)/s(k-1)$ we obtain

$$p(k) = b(k+1)q(k) - b(k)s(k-1),$$

or explicitly

$$a_0(n+1-k) + a_1(n+1) = b(k+1)(n+1-k) - b(k)k.$$

In principle $b(k)$ is still a rational function, but all rational solutions to an equation like this (with p, q and s polynomial) are polynomial themselves. If $b(k) = ck^m + \cdots$ is a polynomial of degree m, we see that the left hand side remains a polynomial of degree 1, whereas the right hand side seems to be of degree $m+1$ (due to terms in $b(k+1)(n+1-k)$ and $b(k)k$, and indeed plugging in b shows that the leading term will be $-2ck^{m+1}$. This means that $m+1 = 1$, so b is a polynomial of degree 0. Thus $b(k) = \beta$ is some constant function in this case. Thus we need to solve

$$a_0(n+1-k) + a_1(n+1) = \beta(n+1-k) - \beta k,$$

where a_0, a_1 and β are still allowed to depend on n. Equating the coefficient of k and the constant term on both sides gives us two homogeneous linear equations in three unknowns:

$$\begin{cases} -a_0 & = -2\beta, \\ a_0(n+1) + a_1(n+1) & = \beta(n+1) \end{cases}$$

As there are more unknowns than equations, we are sure to find a solution, for example $a_1 = 1$, $a_0 = -2$, and $\beta = -1$. Thus we recover $a_0 f(n) + a_1 f(n+1) = -2f(n) + f(n+1) = 0$. We also find the explicit expression for G.

You do not have to do all these calculations by hand, you can just check the final result works, which is much easier. Of course, there remains the part of solving the resulting difference equation, but that is a story for another time. If you want, you can read all about it in the well-written book $A = B$, by Petkovšek, Wilf, and Zeilberger and can be freely downloaded from the web (https://www.math.upenn.edu/~wilf/AeqB.html).

The blockchain revolution

By Niels, employee of OVSoftware

A few years ago, the blockchain suddenly came on the scene; a new technology that interested few people at first. At the time, it was significantly less valuable than it is today. However, companies are now eagerly experimenting to see how blockchain can fit into their organisations. Fundamentally, blockchain is a data structure for digital transactions that are visible to everyone. But how do companies apply this? Niels had the opportunity to get started with blockchain for a government agency.

The basis of blockchain

Perhaps you already know about the possibilities offered by blockchain, and perhaps not. Blockchain technology is an open, distributed and immutable database. In short, this means that the data saved in a block is visible to everyone. The data is not stored in a server; instead, each point in the network has its own copy, and once the data is stored, it can no longer be changed. Of course, this is a very simplified view, but it is essentially the basis of the technology behind blockchain. It's a smart mixture of various familiar technologies, such as cryptography, peer-to-peer networking and consensus theory.

As the name indicates, blockchain is actually a chain of blocks containing data. New data is offered to the network, a new block is created from this data, and the block is then `linked' to the previous block. This is by no means a simple task. Afterwards, the new block is distributed among all of the points in the network. The blockchain network is simply depicted in the illustration below. This clearly shows that all of the points are in contact with one another. Each point possesses its own copy of the blockchain.

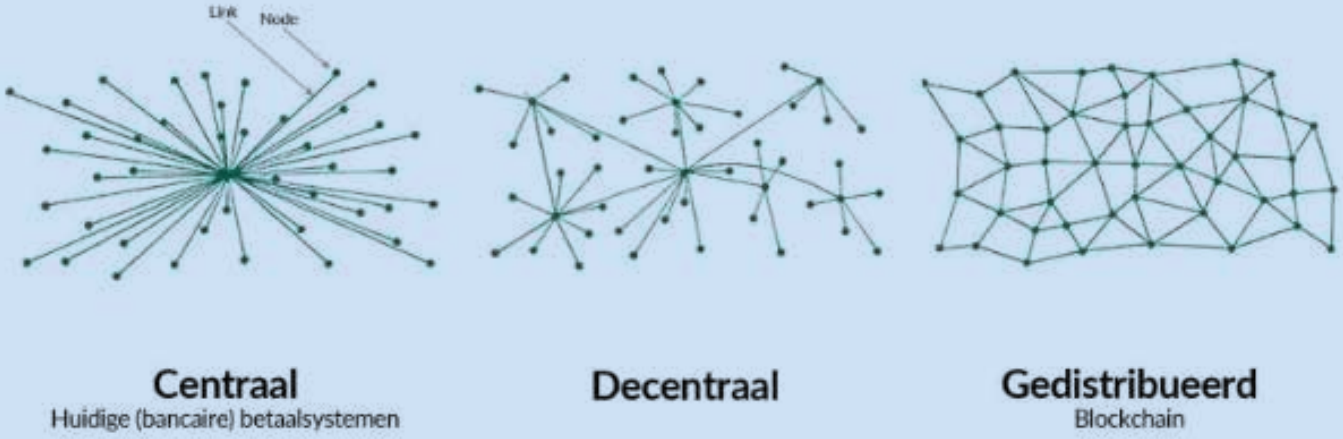
As previously indicated, creating a block is no easy task. There are many requirements associated with crea ting a block, which means that a lot of processing power is necessary to achieve it. When we talk about processing power, we mean entire sports halls full of computers. This processing power is provided by special miners, who are in turn rewarded for creating a new block. This reward can consist of transaction costs that are paid for the storage of the data, or in the form of bitcoins. A lucrative business, the bitcoin has recently been worth around $20,000. However, it is currently decreased to around $9,000 which shows that the business remains unpredictable.

New possibilities

Although it's fantastic of course that the bitcoin is still increasing in value, blockchain is primarily interesting to companies for data storage. Because the data is immutable, blockchain technology makes value exchange possible without a trusted third party being necessary, such as a bank.

In the Netherlands, there is a great deal of experimentation being carried out by various government agencies. Blockchain offers a lot of opportunities. For example, imagine that you would like to purchase a piece of land. Currently, you have to request paperwork from a number of different agencies to prove that you can actually buy that piece of land. This takes a lot of time and

> "Once the data is stored, it can no longer be changed"



money, which nobody finds desirable. This is one of the reasons why there are various pilots running at government agencies. Niels was allowed to work on one such project:

`A while ago, I was allowed to work on a project for a government organisation whose work involves recording a lot of data. At the moment, there are many transactions processed every day which makes blockchain technology very interesting for them. Blockchain makes it possible to capture transactions unequivocally. Partly because of this, they wanted to start up a pilot to investigate how useful blockchain technology is for their domain.

During the project, we created a proof of concept application for a defined use case. In this use case, it had to be possible to request data from the agency. It also had to be possible to call on a service to verify the accuracy of the data.

"We should all work together on this new, cool technology"

Because this service is based on blockchain technology, no one has to pursue this personally. Users can check for themselves whether all of the data is correct, given that this is immutable in the chain.

A website was also created as part of this proof of concept. It's possible to request data on this website. For this, the website approaches a new service that requests the data, writes a checksum on the blockchain and returns the data. The website can now consult the blockchain and retrieve the checksum of the data.

This means that the retrieved data is also checked for correctness. By recording only a checksum of the data, many requests can be added to the blockchain without the blockchain growing too fast.

I worked on this project with a group, and we carried it out in four sprints of a week each. It was a relatively quick and short project, but the result is fantastic. At the moment we're looking at how we can follow up and further develop the project, so it can actually be used in the future. In my opinion, blockchain is a really great, interesting technology and we're going to hear a lot more about it.'

The future of blockchain

As already mentioned, there is a lot of experimentation going on with blockchain. Whether it's for small projects or large ones, people are eager to find out how blockchain can contribute. The question now, of course, is whether or not this trend is going to continue. Is blockchain going to have a large impact on our daily lives in the future, or will the interest in it eventually vanish into thin air? Whatever the outcome, everyone's continuing to experiment with it. And if we can believe what Niels says, we should all be working together on this new, cool technology.

About OVSoftware

OVSoftware was founded by the Oude Velthuis family in 1972. OVSoftware is one of the first software companies in East Netherlands. By constantly innovating, OVSoftware managed to survive all the storms of the rapid hardware developments. That's not easy, but all colleagues have a lot of knowledge about the IT world that we constantly share with each other.

OVSoftware is a company with short lines, which means we have an informal environment. This way we can also guarantee that you will work on a project that you like. Because of the long-term relationships OVSoftware has with their customers, they know how their business and IT environment works. This ensures that they can help their customers from scratch and develop custom software in various market segments.

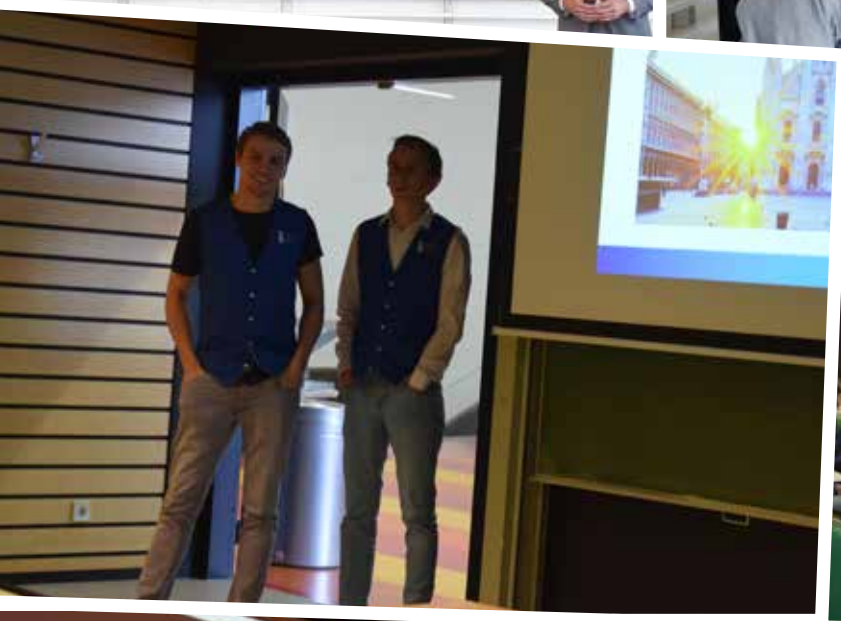
When you are employed by OVSoftware, you have the guarantee to work close to home. OVSoftware has offices in Amersfoort, Apeldoorn, The Hague, Oldenzaal and Munster (D) and focuses on Java and.NET technology.

If you want to see for yourself who OVSoftware is, you can fill in the coffee drink form on their website. Meet them without any obligation!

OVSOFTWARE

Advertorial


Analytics




na, Niccolo [Tech] Hi
ot
l help you today?
na, Niccolo [Tech] How are you?
ot
hank you!
na, Niccolo [Tech] What can you do?
Bot
ort bot, ask me questions!


Dr. Ir. M.C.Veraar

Miscellaneous

Science Trends

Rebecca Glans, Editorial Staff MaCHazine

On this page you will find some brief info on recent scientific breakthroughs or interesting news. Wherein the past issues this could include various subjects, this issue we chose to focus on just one: Artificial Intelligence (AI). Instead of getting a glimpse of different (emerging) trends, we would like to give you more insight in one. Do you prefer the previous approach or do you have another great idea for this page? Feel free to contact us as we would love to hear your ideas!

Our brain's GPS

In 2014 a Norwegian research team gained a Nobel prize for their discovery of how rats track their location. Their brain projects an imaginary hexagonal lattice, with the help of grid cells, onto their surroundings to create an internal coordinate system. These kind of brain nerve cells have also been found in human brains and are suspected to also help in planning paths in between points.

These findings have been tested with AI. In Figure 1 an AI with grid cells and an AI without grid cells had to solve a maze. Afterwards, they had to solve the maze again but with extra paths available (the so called "doors" were open). The AI without grid cells took the same path as before, whereas the AI with grid cells altered its path to reach its destination faster.

This discovery does not only support the mentioned expectation, but also shows how experiments on AI systems could eventually replace animal testing. There are limitations however, as you aren't sure how the AI learns.[2]

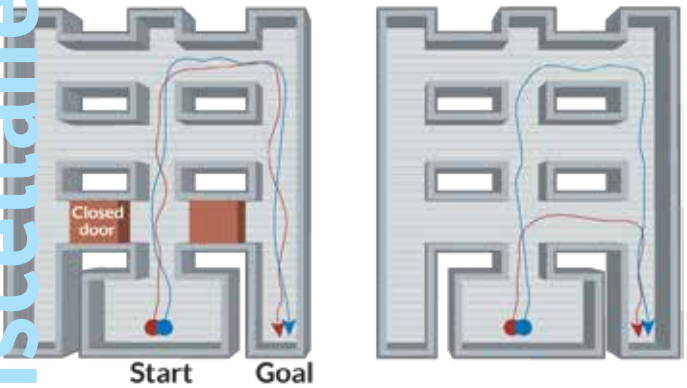


Figure 1: An AI with grid cells (red) and an AI without grid cells (blue) solving a maze (left) and then solving an altered version of the maze (right). [2]

Need some assistance?

One of the numerous fields where robotics could be applied to is healthcare. The tasks such a robot could do are endless and are often very difficult to translate to machine actions. One of these tasks is helping a person put on clothes. How much force should you use? How should the piece of clothing be worn? How do you guide a piece of clothing along the body of a human? The Georgia Institute of Technology helped a robot to learn the first steps in clothing someone: it can successfully and comfortably slide a hospital gown on people's arm. By watching simulations of a robot both successfully and unsuccessfully sliding a gown on a human arm, it learned what a person might physically "feel" when getting dressed. *"People learn new skills using trial and error. We gave the PR2 the same opportunity,"* said Zackory Erickson, the lead Georgia Tech Ph.D. student on the research team. Through the simulations it learned to predict what could happen when using different movements and thus "think ahead" to slide the gown successfully. Instead of using vision, the robot uses its sense of touch to do the task. *"The more robots can understand about us, the more they'll be able to help us,"* Charlie Kemp, an associate professor in the Wallace H. Coulter Department of Biomedical Engineering at Georgia Tech and Emory University and the lead faculty member, said. "By predicting the physical implications of their actions, robots can provide assistance that is safer, more comfortable and more effective."[1]

Don't be evil, Google

AI isn't a topic in research alone. Many (commercial) businesses and governments are nowadays exploring its possibilities. The past months many academics have urged Google to abandon the U.S. Department of Defence (DoD) and their so called project Maven. This project uses AI and machine learning to analyse huge amounts of data to "turn the enormous volume of data available to the DoD into actionable intelligence and insights at speed". It has been reported that Google-employees have resigned because of their involvement in project Maven. Also, thousands of (current) Google-employees have signed a petition to protest against their focus on warfare as they believe it is no place for Google. If you weren't aware of this affair or aren't really informed, it's a good and topical thing to read up on and to determine where you stand in this debate.[3]

References:

[1] Robot teaches itself how to dress people (2018, May 14). Retrieved May 22, 2018, from https://www.sciencedaily.com/releases/2018/05/180514122506.htm

[2] This AI uses the same kind of brain wiring as mammals to navigate (2018, May 9). Retrieved May 22, 2018, from https://www.sciencenews.org/article/AI-uses-mammal-brain-wiring-navigate?tgt=nr

[3] WHAT IS PROJECT MAVEN? GOOGLE URGED TO ABANDON U.S. MILITARY DRONE PROGRAM(2018, may 15). Retrieved May 22 2018, from http://www.newsweek.com/project-maven-google-urged-abandon-work-military-drone-program-926800

Historical Person: Leonhard Euler

Daniël van Gelder, Editorial Staff MaCHazine

Life and career

In the 1700s, it was not uncommon for people who pursued an academic career to be involved in a very wide range of fields such as: philosophy, physics, astronomy or mathematics. Although the combination wouldn't make much sense in our current society, for Leonhard Euler it was what he would be pursuing during his lifetime. Leonhard Euler was a Swiss academic known for many contributions to mathematics and physics. Born in 1707 into a religious family, he was destined for a life in priesthood. However, his life took a different turn when his tutor Johann Bernoulli, another prominent mathematician at the time, discovered his pupil's aptitude for mathematics. At the age of sixteen, Euler had already received a Master of Philosophy, after which Bernoulli urged him to pursue his mathematical ambitions. After finishing his philosophy studies, he pursued theology for a while, as his father wished, on which he graduated in 1724. Now his passion turned more onto the sciences of nature, where he researched the speed of sound, on which he graduated in 1726. By his nineteenth, Leonhard Euler had already mastered three scientific fields.

In 1727, Euler left for Saint-Petersburg, where he spent some years as a professor in physics and mathematics. On invitation of Frederick the Great, he travelled to Berlin to become a member of the Academy there in 1742. However, he later returned to Saint-Petersburg. Throughout his life he suffered from deteriorating eyesight and already in 1738 he had lost one eye because of a disease he had contracted. Later on in his life, his remaining eye started to get worse as well. Surprisingly however, this did not influence his productivity and he continued his academic work as if there was nothing wrong. He used his skills in mathematics and his photographic memory to deal with this. For example: he could recite the entire Aeneid from Virgil from beginning to end. In fact, his productivity is said to have improved over time. In 1775 he published about one article per week on average.

Contributions to science

Although Euler's work was very extensive, covering a lot of topics which were not necessarily related to mathematics, there is no space to cover all the discoveries he made. His collected work spans around 70 parts. Here I would like to pick out a few I found interesting and worth mentioning.

Euler was one of the pioneers in formulating a mathematical notation for a variety of concepts. One of these was the introduction of a notation for a function. We know this as $f(x) =$, which describes a function f applied to an argument x. Some other notations that he introduced or popularized: e as the base of the natural logarithm (also known as Euler's number), the modern notation of goniometric functions, the letter Σ for describing summations, i for imaginary numbers and π, although he was not the one to introduce this notation, for the ratio between the circumference of a circle and its diameter. So we actually have many notations to thank Euler for in current mathematics.

Due to the influence of the Bernoulli family on his education, Euler spent a lot of time on calculus, as the Bernoulli family was at the forefront of development of infinitesimal calculus. Euler often used $powerseries$ in his proofs, for example to express terms like:

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \cdots + \frac{x^n}{n!} = \sum_{n=0}^{\infty} \frac{x^n}{n!} \tag{1}$$

Euler also presented a solution to the *Seven Bridges of Königsberg* problem which many of us will know. It is a problem in graph theory where parts of a city which are divided by a river are connected by seven bridges. The problem is to determine whether it is possible to traverse all bridges in such a way that each bridge is visited once. Euler was the first to present a solution which rules out the possibility of a path that traverses all bridges once. This solution presented a new concept in graph theory: an $Eulerianpath$. An Eulerian path is a trail in a finite graph which visits every edge exactly once.

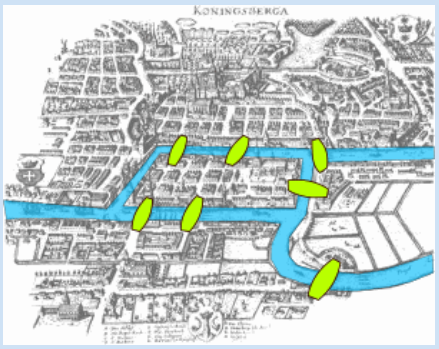


Figure 1: Illustration of the Seven Bridges of Knigsberg problem

Death and commemoration

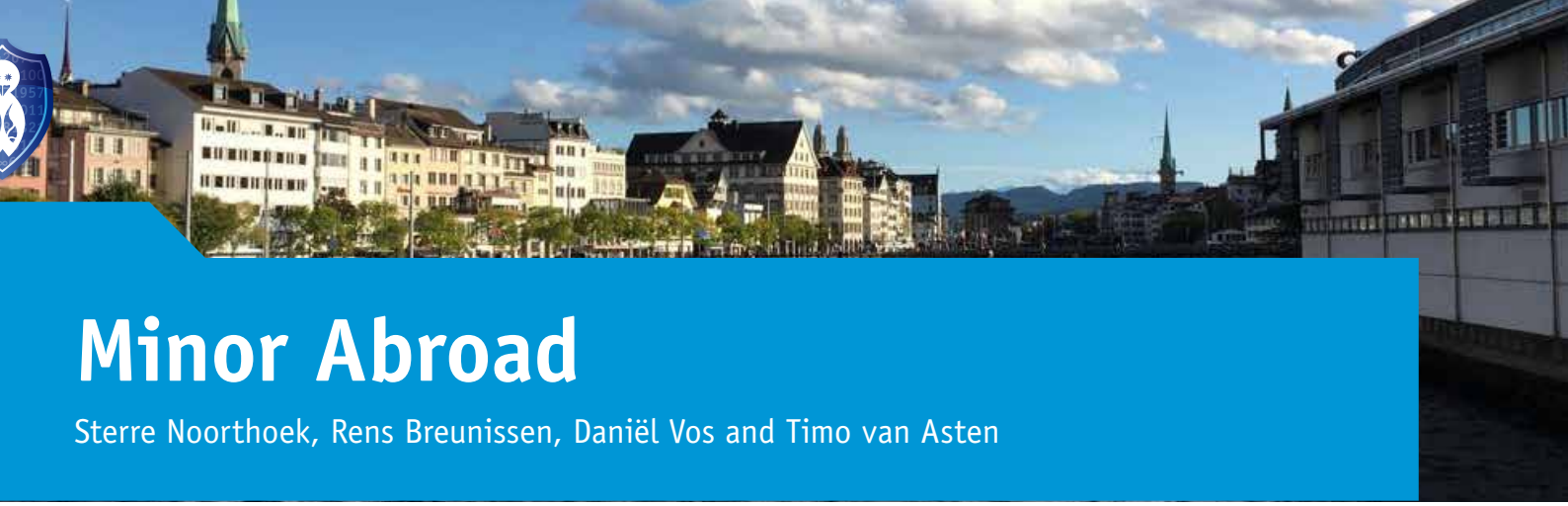
Leonhard Euler's phenomenal scientific career came to an end on September 18th, 1783 when he died of a stroke, aged 76. In a tribute speech in name of the French Academy, the mathematician and philosopher Marquis de Condorcet wrote: "*il cessa de calculer et de vivre...*" (he ceased to calculate and to live).

On his 306th birthday, in 2013, Google changed its logo to a doodle commemorating Euler and his discoveries. The asteroid 2002 Euler was named in his honor. However, he is best remembered by the many discoveries in his name which have formed a large part of the fundamentals of modern day mathematics.

References

[1] ttps://historiek.net/google-eert-wiskundige-leonhard-euler-met-doodle/21993/

[2] ttps://en.wikipedia.org/wiki/Leonhard_Euler



Minor Abroad

Sterre Noorthoek, Rens Breunissen, Daniël Vos and Timo van Asten

For this edition of the MaCHazine, we asked some students who went abroad during their minor to share their experiences. Sterre and Rens both went to Australia, Daniël went to Switzerland and Timo went to the United States. Enjoy reading their stories!

Sterre Noorthoek - Melbourne

Last year I did my minor in Melbourne! Before flying to the other side of the world, the necessary paperwork needed to be completed. I even had to write a small essay to convince the Australian Government that I had no intention of staying after my visa would expire. They do know how to raise expectations!

On the 11th of July (and the 12th actually, yes, long journey), I flew to Australia. I had planned to start looking for housing on site. Luckily, I could soon trade my hostel for a house with American, English and Dutch exchange students (nine women and one man). We lived in Fitzroy, the super hipster suburb close to the CBD (city center) and campus, where it seemed like everyone was vegan, did yoga and wore vintage clothing. With countless cafés, clubs and restaurants around, there always was something to do.

To my surprise, the University of Melbourne also offered many activities. Every week there would be a band, a free barbecue, and food markets. Not to forget the hundreds of student clubs. I joined the cheese club, the wine club, and the Street Workout and Calisthenics Club. Sports that I played were canoe polo and cricket.

Naturally, I also came to the campus to study! I followed fun and interesting courses in information systems, cyber security, interaction design, and marketing. New to me were the many essays I had to write and presentations I had to give. In addition, there was a group project consisting of both Australian and international students for every course. Working on these projects was always a lot of fun and a nice way get to know fellow students better.

My exchange came to an end with my last exam in November. Though it was not time to head home yet! I travelled through Tasmania, New Zealand and the east coast of Australia for two months, which was truly beautiful. After being away from home for over six months it was time to go back with this insightful experience in mind.

Rens Breunissen – Melbourne

For my minor abroad, I spent a semester studying at RMIT university in Melbourne, Australia. I wanted to try something new, so instead of taking courses related to my major, Applied Mathematics, I decided to take some courses in civil engineering instead.

RMIT is a technical university, so very similar to TU Delft. The thing that shocked me most was the lack of activities in and around the university. Whereas we have many study clubs and student clubs associated with the TU Delft, there are very few of these clubs at RMIT and most people just come in for lectures and head right back home again. This means that you have to put in more of an effort to get your own social group, but when you do it's definitely worth it. The people are probably my favorite thing about Australia. Everyone is easy going, very open and they love to meet new people.

I ended up joining a rowing club myself, as I am usually a pretty active rower, and spent most of my time there. The atmosphere around the club was great, and you get some great views as you can row right through the heart of the city!

The biggest downside to living in Australia would be the cost of living. Living in Australia is more expensive than in the Netherlands, so be sure to save up if you plan on going there. If you're not afraid to spend the first week in a hostel, you should look for a room while you're there to save some money on rent. It's definitely worth it to travel around if you have the time and money. I spent a month traveling down the east coast, but I would advise to take at least twice as long if you want to do the country any justice, because there is so much to see.

Daniël Vos - Zürich

Hey, I'm Daniël and I am in my third year of the Computer Science and Engineering Bachelor. In high school I followed a bilingual program, so that I would easily be able to go abroad to study. That's why, when it was time to do a minor, I naturally chose to do it abroad and I ended up studying a semester at ETH Zürich.

The curriculum at ETH is quite different from TU Delft as courses take 4 months and exams take place after a short holiday, or rather, a study break. The workload was higher than in Delft and I soon learned that this was the reason why most students took a maximum of 25 EC each semester. The courses, however, were taught really well and the professors were motivating and inspiring.

ETH offers students free access to the gym and group classes which I gratefully used. Housing was awesome, as I lived in a building filled with only exchange students. Needless to say, every day was a party. Also, in the weekends there was plenty to do. ESN Zürich, the network of exchange students in Zürich, organized a lot of trips and it was common for students to organize trips themselves too. The second largest formal ball of Europe and a house party from someone I didn't know in a town I didn't know are two experiences I will never forget.

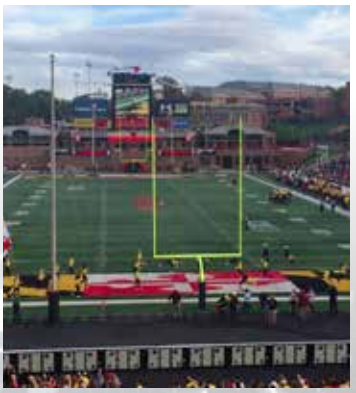
A minor abroad at ETH means hard work, but it is really worth it. Getting to know people from all over the world and discovering places while doing a minor is something you can only really do abroad.

Timo van Asten - Maryland

Hello reader, I'm Timo, and to start off I would like to say: if you get the chance to do so, go abroad for your minor. It was an experience I will never forget, and I met many people that gave me so much energy, love and friendship. For my minor I went to the University of Maryland; a university close to Washington DC. It was a decision I took somewhat impulsively, but I am so glad I made the decision to go. A friend texted me that the deadline for a minor abroad was 2 days later, and I did not pay attention to this deadline at all. In the short time that I had, I wrote a motivation letter and did some research. The University of Maryland ended up in my top three. Four days later I received an email that I was selected to go, and 7 months later I got onto a plane to the US.

It is amazing how refreshing it is to get away from the daily routine. Everything you see is new. Everyone you meet is new. American college life is so different from ours, which you notice in a few things. The whole campus is like a big town, where students are the only citizens, and this makes it ridiculously easy to make new friends. There is a party around every corner (yes, also the frat party's you see in the movies). You don't cook or do the dishes, but there is one big dining hall where all the students eat, and all of this is done for you. About every two weeks there is an American football match which you can go to free of charge. Everyone dresses up in UMD clothing and students have drinks together from the back of their car, called 'tailgating'. It is one of the most exciting days on campus called 'game day' and it is so cool to experience this yourself. I got to see Washington DC, New York, Chicago and Miami.

… and of course, you also study, but you don't even notice that you are.



Studying abroad was just what I needed to bring back my joy in studying. It has changed me as a person and living in another country for 4 months makes you appreciate things about your own country you didn't even know were special. Just go!

Mathematics Puzzle: Nonogram

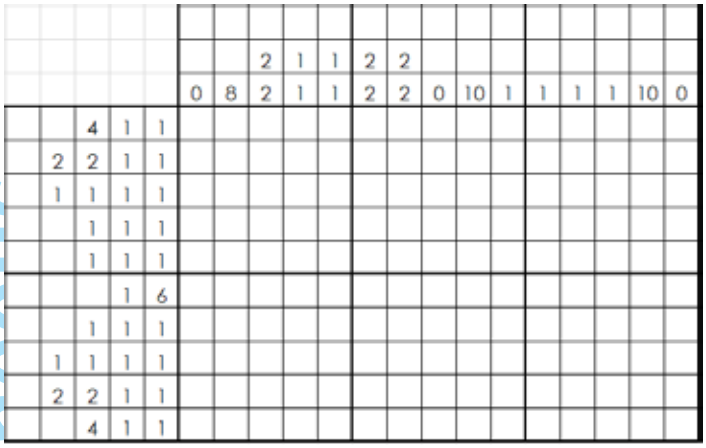
Eva Slingerland, Editorial Staff MaCHazine

This puzzle is also known by some other names like Paint by Numbers, Griddler, Pixel Puzzle, Japanese Crossword, Japanese Puzzle, Kare Karala, Oekaki-Mate, Picture Logic and so on.

It works like this: the numbers on the left side of each row indicate the order and length of consecutive unconnected black segments. For example, if it states '4 1 1`, it means that from left to right, this row contains a black segment of length 4, followed by a black segment of length 1, which is again followed by a black segment of length 1. The black segments are separated by one or more white segments of unknown lengths. There could be a white segment of unknown length before the first segment or after the last segment. The same goes for the numbers above each column. It is useful to also mark the segments of which you are sure they will be white, by for example a '-`.

For example, after solving the first puzzle, the first row will look like this:

Let´s start off with puzzle 1:



Solutions from the last issue

Average

When imagining this list, we notice that the amount of numbers that end on a 1 is the same as the amount ending on a 2, and the same as the amount ending on a 3, and the same amount of numbers ending on a 4. So a quarter of the numbers ends on a 1, a quarter on a 2, a quarter on a 3 and a quarter on a 4. On average, the last number is

$$\frac{1+2+3+4}{4} = 2.5$$

Exactly the same holds for the other positions. So the average for the dozens is 25, for the hundreds it's 250 and for the thousands it's 2500.
This means the average of all the numbers in the list is:

$$2500 + 250 + 25 + 2.5 = 2777.5$$

Biggest number

From $a+b+1 = b+c-2$ follows $a+1 = c-2$, so $c = a+3$. In the same way, it follows from $e+a+5 = d+e-4$ that $d = a+9$. And $c+d+3 = d+e-4$ leads to $e = c+7 = a+3+7 = a+10$. And $b+c-2 = c+d+3$ leads to $b = d+5 = a+9+5 = a+14$. So the numbers a, b, c, d, e are $a, a+14, a+3, a+9, a+10$, respectively. This means b is the biggest.

Division

We try to find a pattern for dividing 1, 11, 111, 1111, and so on, by 37:

$$1 = 0*37+1$$
$$11 = 0*37+11$$
$$111 = 3*37+0$$
$$1111 = 30*37+1$$
$$11111 = 300*37+11$$
$$111111 = 3003*37+0$$

Here we see a pattern, which continues! (You can try this yourself). Apparently, the remainder is zero when the number consists of a triple of ones: 3 ones, 6 ones, 9 ones, and so on. Because $2010 = 670*3$, we know that 2010 is in the sequence of numbers that has remainder 0. This means that the number with 2011 ones has remainder 1, according to the pattern.

Puzzle 2 is much larger! Take some time to create the next image.

													3						3																
								1	1	2	5	3	2						2	3	5	2	1	1						2	2	3			
						4	7	10	15	3	11	2	11	4	4	8	4	4	11	2	11	3	15	10	7	4			1	1	1	2	9	7	4
						3	3	3	3	15	1	15	3	16	18	15	18	16	3	15	1	15	3	3	3	3	3	3	3	3	3	3	3	3	3
				4	4																														
			5	5	4																														
				11	6																														
				13	4																														
		2	5	2	3																														
		2	3	2	3																														
1	2	1	2	1	3																														
2	2	1	2	2	2																														
		3	1	3	3																														
		5	3	5	4																														
				10	10																														
				10	10																														
					21																														
					19																														
					19																														
					17																														
					17																														
					15																														
					15																														
					15																														
					13																														
	5	1	5	1	14																														
	5	1	5	1	14																														
					30																														
					9																														
					7																														
					3																														
					0																														
					0																														
					0																														

Miscellaneous

Of risk, fences, and unavoidable falls

Pasquale Cirillo, Assistant Professor of Applied Probability

Rule 1 of Risk Management:

it is when you finally feel safe that you are not.

Rules? Too many rules!

We live in exceptional times, in which what was considered the norm is no longer the norm. On the markets we can observe negative interest rates, and central banks are implementing monetary policies never seen before. Giant companies like Lehman Brothers disappear in a heartbeat, while new ones emerge with an app. Everyday we give up some personal liberty, and our privacy, in the hope to protect ourselves from terrorism.

As observed by philosopher Giorgio Agamben [1], we live in a continuous "state of exception", in an *Ausnahmezustand* according to another scholar, Carl Schmitt [5]. Exceptional situations, or at least perceived as such, make Governments suspend, repeal or amend the current legislation, proposing new restrictive rules, new constraints, with the alleged primary goal of solving the problem. In the face of some events, we feel lost, we are afraid, and we lose our traditional reference points. It is exactly then that politicians take the opportunity to grant us some new feeling of safety, to bring things back to order. And to expand their power.

The number of examples is large and growing. We can recall the *Patriot Act* in the US, after 9/11, recently partially amended by the *Freedom Act*, echoed by several laws that have been approved in Europe in the last few years, up to the famous state of emergency declared in France. And in the financial world, after the 2008 and 2012 crises, we can cite the long overwhelming sequence of rules created by the Basel Committee on Banking Supervision, collected under the name of Basel II and III, whose goal is to regulate the existence of banks and financial institutions, in order to reduce both the idiosyncratic (at the bank level) and the systemic (at the economy level) risk of losses and defaults.

Basel rules, whose number of pages is greatly larger than that of the Bible and the Quran stacked together, are a clear example of the excess of regulations we are experiencing today. Regulations aiming at reducing risk, at giving us safety and efficient markets, but that, everyday, are showing their true face. They are a problematic burden, an often useless cost, a heavy lid over a pan with boiling water, with steam and pressure mounting, up to the final explosion. A driver, not a cure, for systemic risk.

The Fence Paradox

How may rules conceived to fight risk actually increase risk? The best answer is given by what I like to call the *fence paradox*, which shows how some protections, some rules, only give a false sense of safety, while actually hiding risk, also modifying our risk perception.

Have a look at Figure 1, box A, where two friends are just on the edge of a canyon, looking down at the end of the precipice. We can imagine they are in Arizona, and they look down to admire the beautiful Colorado river turning red because of the colored sands you can find over there. Unfortunately, to profit from such a beautiful view, you have to lean out and risk.

The two friends have different attitudes. Lehman is a risk taker, he is bold, and he does not want to get his new t-shirt dirty. He feels safe on his legs: nothing bad can happen! John is different, very cautious, Lehman calls him a coward, as he does not like risks. He feels safer if he lies down on his belly and just leans out with his head.

Unfortunately–the reason is unknown to us, maybe a headache or a distraction–Lehman loses balance and falls down. In box B we see the tragedy happening.

When you take risks, negative consequences are always around the corner, it is unavoidable. Risk is indeed defined as the probability or threat of a negative occurrence, caused by external or internal vulnerabilities, which you may try to mitigate and hedge, often only partially, given the limited resources available [2]. Notice that the concept of negative occurrence is often very subjective: losing 1000 euros is probably not the same for you and Bill Gates. And also the same definition of probability may be subjective [3]–actually always is[1], but we do not have the space to go into the details here.

As one of the sacred texts of Hinduism, the $Bhagavadgītā$, points out: risk is a human thing, there exists no human activity without risk, and there is no risk without humans. You can decide not to leave your house, to do nothing and be safe, but there is always a chance, maybe small but not null, that the roof falls down. And what would be the risk of a earthquake if nobody could get hurt? How does our risk perception about the same earthquake change, if it hits us directly, or if it happens on the opposite side of the globe?

But we do not like risk. Despite risk being an intrinsic characteristic of our existence, we would like to live without it. Authorities know that and, to avoid new tragedies, and even more to make people feel safer (remember that, in a democracy, people vote sooner or later), they build fences, as in box C of Figure 1.

Probably you do not know, but over there Lehman's death was the first and only death by falling. A tragedy, as all deaths, but we cannot deny that Lehman had taken an excessive risk. Nobody had had a similar reckless behaviour before.

In any case, the new fence looks like a success. Several tourists that were afraid of falling down and avoided the canyon are now coming in groups. The fence allows them to look down with safety. Authorities are praised for the clever decision of building a fence, to the point that local politicians swear they will "do whatever it takes[2]" to make the canyon a safe place for everyone. Better and nicer fences will be built, if necessary.

Rumours spread: the canyon is now safe! More and more tourists arrive. One day, there are so many people that the queue is extremely long. Ok, with so many people looking down, it can feel a little crowded, but there is no need to push! Everyone can enjoy the beautiful sight, leaning against the fence. If you have time, in box D there is still some little space for you to join the

[1] It is not necessary to follow de Finetti's approach to see that probability, in real life, is bound to be subjective. Probability is the way in which WE describe chance, it is our logic of the uncertain. If you choose a given definition of probability you are making a subjective choice. Moreover, assume for a second to be a "frequentist", so that for you the probability of an event can be estimated using the relative frequency of that event in a very long series of trials. Well, unless you restrict your attention to thought experiments, the way in which you collect (and process) data can never be fully objective.

[2] This is a secret quote. A little quiz for you.

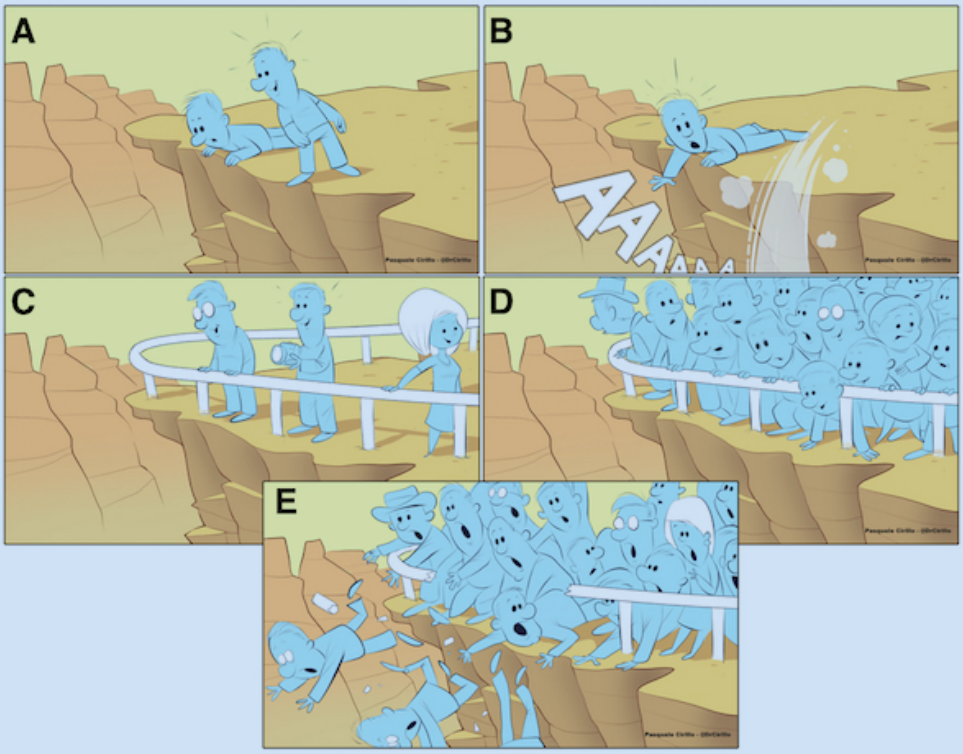


Figure 1: The fence paradox, or when fences are not as safe as you might think.

group.
All of a sudden, the unexpected (?) happens. In box E, because of the excessive number of people pushing, the fence breaks down, and 23 persons die. A tragedy, something never seen before. An unforeseeable extreme: a black swan[3]!
How can it be? What about the fence? It is the fault of the authorities! No, it was the fault of the reckless people. We need a stronger fence! No, we need to forbid any entrance to the canyon.
They felt safe. They were not.

Let's accept risk!

Many rules work exactly as the fence of this little story. They give us a false sense of security. We want to avoid risk even when it is not possible, and we build fences, walls, protections.
When we see fences, we feel safe, and we tend to forget the risks fences are supposed to take care of. We start being less rational, if not irrational. Nothing bad can happen if there is a fence!
But it is exactly when we start feeling very safe that actually we are not. If we do not accept the human nature of risk, if we do not understand that risk management is meant to limit unnecessary risks, to mitigate the unavoidable ones, but not to prevent all risks, we are just preparing the next tragedy, the next crisis, the next state of exception.
Rules are needed, as we need good mathematical models to understand and try to hedge risk. But overconfidence in rules and models[4] is dangerous, as it is dangerous to create too many rules, which only create confusion, and are often subject to the so-called heterogony of ends, generating unintended consequences.
The best way to protect ourselves from risk is to study it, sure, but also to accept it, to understand that we always need to be cautious, because our fences may break, our models may be wrong, our rules may prove insufficient. And we do not need anything too sophisticated to bring back some old grandma's good common sense [6].
Very often, what we think is a black swan is nothing new. It was always there, in front of us, but we had it painted in white.
If you are willing to read more on this topic, I suggest the nice work by Greg Ip [4], and my forthcoming book [7], where the fence paradox is also presented mathematically, in more rigorous terms, as formalized risk homeostasis.

[3] Unfortunately, the expression "black swan" is often misused and abused. Extremes and rare events are not always black swans. Our fence breaking down is NOT a black swan! For the correct definition, I refer to the original, important book by Nassim N. Taleb [6].

[4] Especially when those models are built on the improbable assumptions of normality, perfect rationality and friends.

References

[1] Giorgio Agamben (2008). State of Exception. Translation of the original (2005). University of Chicago Press.

[2] Peter L. Bernstein (1996). Against the Gods: the remarkable story of risk. Wiley.

[3] Donald Gillies (2000). Philosophical Theories of Probability. Routledge.

[4] Greg Ip (2015). Foolproof: why safety can be dangerous and how danger makes us safe. Little, Brown and Company.

[5] Carl Schmitt (2007). The Concept of the Political. Translation of the second edition of the original (1963). University of Chicago Press.

[6] Nassim N. Taleb (2007). The Black Swan: the impact of the highly improbable. Random House.

[7] Nassim N. Taleb, Pasquale Cirillo (2018, forthcoming). Fat Tails: the logic and the statistics of extremes. Penguin Books.

Computer Science Puzzle

Beryl Sicam, Art Director MaCHazine

On the right, you can find the solution to the puzzle we shared last week. Having last issue's puzzle in mind, this one should be *fairly* easy. Provide declarations for i and j that turn this loop into an infinite loop:

```
while (i <= j && j <= i && i != j) {
}
```

The answer will be published in the next MaCHazine!

Solutions from last issue

In the previous issue we asked you if you knew what this program would print, without actually running it on your computer. With an extra challenge if you also knew **why** this result is printed.

```
Integer a = 42;
Integer b = 42;
System.out.println(a == b);
Integer c = 666;
Integer d = 666;
System.out.println(c == d);
```

The program would print

```
true
false
```

There is a thing in Java called Autoboxing and Unboxing. Autoboxing converts int objects to Integer, double to Double, and so on. Unboxing goes the other way around. However, this is only done automatically for Integers between -128 and 127. So the first comparison is seen as

```
int 42 == int 42
```

which is true. The second one won't be automatically unboxed and will be seen as

```
new Integer(666) == new Integer(666)
```

which is comparing two different objects and thus results to false.

References

[1] Java puzzlers: Traps, Pitfalls and Corner Cases, Joshua Bloch and Neal Gafter

Wiskunde
Informatica
Studievereniging

'Christiaan
Huygens'

Become a mentor

Registration will open
1 May 12.49 h

Freshmen Weekend 10-12 aug
wisv.ch/mentor

- ☐ Educational
- ☐ Career
- ☑ Social

Calendar

Enjoy the holidays!

The holidays are almost there! Enjoy sitting back in the sun without worrying about studying for two months. The Board and the MaCHazine committee wish eveyone happy holidays!

Freshmen Weekend

Every year at the beginning of the academic year, an introduction weekend for freshmen students is organized. During this weekend, the new freshmen will get to know their fellow students in an informal way. Several teachers, faculty staff and academic counselors will be present this weekend, so that they can already get familiar with some of them.

July

2-6 Exams

6 Last day of Acedemic year

August

10-12 Freshmen Weekend

31 CoH Alumni barbecue

In memoriam

Professor Doctor Johannes Michael (Jan) Aarts

Honorary Chairman of W.I.S.V. `Christiaan Huygens'

April 17, 1938 - June 14, 2018


A VITTORIO EMANUELE II I MILANESI