

MACHAZINE

Volume 17 - Issue 3
March 2013

Wiskunde
Informatica
Studievereniging



'Christiaan
Huygens'

STIP, ORAS, LIJST BÈTA, VSSD

External Boards

BEST LECTURER OF THE YEAR

Joost de Groot

WINTER WONDERWEEK

WiFi

HISTORICAL PERSON

Edsger Dijkstra

CONTAINING:

CURRENT AFFAIRS | ASSOCIATION | COMPUTER SCIENCE | MATHEMATICS | MISCELLANEOUS



Starting your career?

Visit the UL Transaction Security in-house day!
31 May 2013 (Friday) 12:15 - 17:30

You are invited!

Join our in-house day on Friday, 31st of May 2013 at our office in Leiden.

We will start with lunch, followed by informative presentations and we will work on an interactive case. Afterwards, we will have drinks together.

Want to participate?

Please send your CV to ul.ts.jobs@ul.com with a reference to the in-house day before the 30th of May 2013 (Thurs) 12pm.

Looking forward to meeting you at the in-house day!



UL TS is part of UL
(formerly known as Collis)



content & colophon

MACHAZINE

is a publication of

W.I.S.V. 'Christiaan Huygens'

Editor in Chief

Peter Pul

Editorial staff

Max de Groot, Derk-Jan Karrenbeld, Friso Abcouwer, Kirsten Koolstra, Claudia Wagenaar, Ralf Nieuwenhuizen, Rebecca Jacobs, Pieter Hamete

Art Directors

Peter Pul, Bojana Dumeljic, Hylke Visser, Floris Verborg, Cathaline Meloen

Contact address

Mekelweg 4, 2628 CD Delft
E: machazine@ch.tudelft.nl
T: 015-2782532

Concept and design

G2O Kesteren

Publisher

DeltaHage bv

Cover

Tapis de Fleurs by Luca Burriesci

Cooperating on this issue:

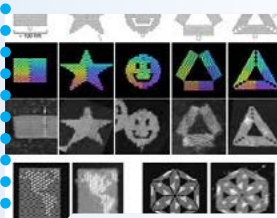
Arie Troebel, Romke Rozendaal, Prof.dr. C. Witteveen, Marieke van der Tuin, Anne Viruly, Eva Nieuwenhuis, Tristan Steegman, Fokke de Groot, Max de Groot, Joost de Groot, Kasper Kooijman, Rebecca Jacobs, Martijn de Jong, Tomas Klos, Vlad Vergu, Noeska Smit, Mark Veraar, Aad Vijn, Jacob de Zoete, Johan Dubbeldam, Margriet Nieuwenhuis, Sonja Cox, Stijn Frima, Frank Tabak

Terms and Conditions

The MaChazine-committee and the Board are responsible for the content within this MaChazine, in such terms that the opinion of a writer is not (necessarily) a reflection of the opinion of the committee or association. All rights reserved. No part of this publication may be reproduced, stored in a (retrieval) system or transmitted in any form or any means, electronic, mechanical, photocopying, recording, scanning or otherwise, without the prior written permission of the committee or association.

Advertiser index

IMC Financial Markets	Outside Cover
UL Transaction Security	Inside Cover Front
ASML	Inside Cover Back
TamTam	15
DSW	24-25
FoxIT	32-33
KPMG	38-39



Current Affairs

Editorial, Agenda and Twitter	2
Arie Troebel	3
From The Board	4
Prof.dr. C. Witteveen - Misverstanden	5
FSR	6
STIP	7
ORAS	8
Lijst Bèta	8
VSSD	9
TU Delft Nieuws	10

Association

Docent van het Jaar	11
Wintersport '13	12
CH Kerstdiner '12	13

Computer Science

A Really Fast GPU Solver for Real-time Computation of Interactive Waves	16
Enumeration and Design of Weighted Voting Games	18
LabBack: Extendible Robust Automatic Assessment of Student Programs	20
Inside the INSIGHT Lab	22
Java Puzzlers	26

Mathematics

Wiskundepuzzel	27
Het Keuzeaxioma	28
The combined evidential value of forensic DNA profiles	30
Modelleren in de nanofysica	34
De verblijftijd van water in een meer met behulp van een deeltjesmodel	36

Miscellaneous

Mijn Uiltje - Alumnus Sonja Cox	40
Interdisciplinary project SimBethel	42
Historical Person: Edsger W. Dijkstra	44

Hylke Visser

In software development you often have the same problem. You can choose to use a new, fancy programming language, which is extremely elegant, but the documentation is still quite disappointing. So are you still going to use that language knowing that you'll have to find out a lot by yourself or are you going to write less elegant code in an older programming language that is much better documented, but also much more devious?

A really good example of problems with cutting edge technology can be found on the Internet. With all those new features in HTML5 and CSS3 every web developer wants to switch to this new technology. But unfortunately, many large companies and organizations still use older Internet browsers that don't support these features yet. So what should you do now? You could use some JavaScript hacks to get at least some things working, but this still doesn't fix the problem.

Well, we are all students at one of the best technical universities in the world. And because we are the guys that are going to create innovative products in the near future we should, in my opinion, use the newest and most cutting edge technology as much as possible. We shouldn't try to get our websites working in some old version of Internet Explorer on Windows XP, just to make our products look good on my granddads computer. We should use our imagination, use the full power of this new technology and with that, we can build amazing things. Don't be afraid to create something entirely different from the things that already exist. We're talking about the future, the things we create for the future should be different from the things we have now. That's the whole point of innovation. 🙋

March

April

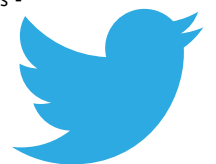
May

Silver for Peter

Pul, thank you for all your hard work for our study association but especially for the MaCHazine. Enjoy reading your 25th edition!

- Your fellow MaCHazine-members -

Twitter



@tomasklos

Assume set P of r primes. Let $x = p_1 \cdot \dots \cdot p_r + 1 > p_r$. If x is prime, $x \notin P$; else, the r primes don't divide x , so for a prime divisor d of x : $d \notin P$. QED

@htonino

TU Delft gaat MOOC's aanbieden op edX, een platform dat opgericht is door o.a. MIT en Harvard. <http://wisv.ch/tudelftmooc>

@eelcovisser

RT @TI1220 Version 36 of WebLab released <http://wisv.ch/weblabv36>; WebLab now open for new course requests.



De Regel van Cramer,of???

Arie Troebel

Lief, lief schatje, in het altijd zo mooie Delft. Ik moet zeggen jullie Arie is druk, druk, druk onderweg van de ene lezing naar de andere lezing, Ik hoop dat alles goed gaat in Ter-Weksel, omdat de heer G. Flater daar momenteel vanwege mijn afwezigheid met de scepter zwaait. Dat zie ik verder wel als ik daar weer terug ben. Maar goed, ik dwaal af. Laat ik het hebben over de 'Regel van Cramer'. Deze jongen leefde van 1704 tot 1752, en was toch wel een begaafd wiskundige (voor alle duidelijkheid: natuurlijk niet van het kaliber Arie). Om deze reden ben ik me gaan verdiepen in het 'waarom' achter de 'Regel van Cramer'. Ik kan veel hebben, maar om nu de 'Regel van Troebel' te plaatsen gaat mij even te ver. Bovendien zouden het vele 'Regels van Troebel' worden.

Meneer Cramer toonde onder meer aan dat een kromme van de n-de graad in het algemene geval wordt bepaald door precies

$$\frac{n * (n + 3)}{2}$$

punten! Tja, daar gaat mijn wetenschappelijke schoen van wringen: hoe en waarom kan dit? Om dit te onderzoeken reisde ik af naar de universiteit van Geneve waar dit heerschap toentertijd op de loonlijst stond. Bij mijn aankomst werd ik met alle egards behandeld, omdat de wetenschapper niet zomaar elke donderdag in hun catacomben verbleef. Even later begon de ellende toen ik vragen stelde over de 'Regel van Cramer'. Men begreep mij niet of ze vertrokken in allerijl. Dit versterkte mijn argwaan des te meer.

Via slinkse wegen (zoals alleen jullie Arie kan) ben ik in het bezit gekomen van het heiligste der heiligen van de universiteit van Geneve, namelijk de sleutel van de trésor afdeling. Hierdoor had ik inzage in de boeken, dus ook die van toen onze Gabriel Cramer nog staande een biertje kon drinken.

Ik sloeg in dat duistere hok de boeken open en verbaasde me keer op keer. Er bleek dus geschreven te staan dat dit heerschap een wel heel erg reislustig type was! Dit alleen zegt nog niets, maar de personen waar deze rakker mee sprak waren niet de minste. Ik zal een paar namen noemen die ik heb gevonden, hou jullie vast lieve lezers.

Meneer Cramer had onder andere een onderonsje met J. Bernoulli, L. Euler, E. Halley, J. Stirling, A. de Moire, P. de Maupertuis, A. Clairaut' en nog vele andere namen. Het zweet brak mij uit en mijn gedachten gingen onwillekeurig naar 'deals', of laten we zeggen 'afspraken', in het wetenschappelijke gebeuren!

Stel je voor dat meneer Cramer tegen andere wetenschappers gezegd heeft: "Laten we de 'Regel van Cramer' met zijn allen in de boeken zetten. Als jullie dan met een leuke naam komen voor een stelling die je hebt uitgevonden, dan neem ik ook dat zonder morren aan!".

Jullie kunnen begrijpen dat ik op dat moment geen enkele historische wetenschapper meer kon vertrouwen. Bibberend liep ik naar buiten, de vrieskoude trotserend, om weer tot rust te komen.

"Arie jongen, je leest teveel fictieve boeken. Houd eens op met die complot theorieën" zei ik tegen mijzelf. Hierna was ik over de streep getrokken. Resoluut stapte ik het eerste beste café binnen, bestelde daar een goed glas wijn en liet alles langzaam van me afglijden. 🍷



From The Board

Romke Rozendaal

Time rushes by, activities accumulate and sleep is nowhere to be found.

That's probably the answer you will get when you ask a random board

member how the life as a board member is. And although I am the last

person to deny that this is true, I would like to tell you a little bit about

the normal workweek of a board member and in my case, the treasurer.

Monday

As usual the week kicks off with our weekly board meeting. With the six of us we discuss all matter, evaluate the finished activities and look ahead to the upcoming week. After the first Monday morning grumpiness slowly I get more and more cheerful. Sitting behind my computer clearing up my mailbox from all the e-mails I received in the weekend and at the same time F5-ing my ING to see whether all our debtors have already paid their bills. In the afternoon I take the train to Amsterdam to one of our partners to talk about the implementation of our partnership in the upcoming year. Once back in Delft I hurry to the football pitch to aid the CH owls in their victory against the TG Tigers. Doing one more victory beer and then it's off to another meeting, this time with one of the committees I assist. Afterwards it's finally time to go to bed.

Tuesday

Unfortunately it's my duty to be the one to open our community space. So getting out of bed early and preparing the coffee and tea for our early birds is my duty. At 9.00 the rest of my fellow board members arrive and a few minutes later we start the meeting with the Board of Acquisitioners. The three of us, responsible for a great part of the incoming money, catch up on each other's progress and make sure all our noses are pointing in the same direction. In the lunch break I have another meeting with the Dies-commission. These awesome fellas are making the most important week of the year even more awesome. After that it's time to keep a warm relationship with the companies I'm responsible for. With my feet on the desk I call some partners and make sure everything is OK and also try to sell them some of our services. In the evening I have a drink with a study association for mathematics outside of Delft.

Wednesday

This day I'm responsible for the tidiness of our room. In the morning I take a detour along the supermarket to provide my fellow board members from a refreshing lunch. In the morning I have got the time to work on one of my personal projects, transferring our accounting system from one kind of software to the other. Also at 11.00 sharp I clean all of the mugs where our members can drink their free coffee and tea out off. In the lunch I have a meeting with the

AkCie, a commission for first year students who organize the ADSL and some lunches. After cleaning up our area it's of course /pub time. Taking a drink and getting some rest while chatting with some other CH members.

Thursday

In the morning I have nothing planned, so it's time to update the bookkeeping. Filling in some invoices, paying a couple of creditors and sending a few invoices. This lunch break we have a lecture about Mathematics and we are responsible for the lunch. So the rest of my morning I fill with stuffing sandwiches. In the evening it's time to attend to a ball of another study association together with the rest of my board and their dates (read girlfriends).

Mon	Tue	Wed	Thu	Fri
06:00 - 07:00 07:00 - 08:00 08:00 - 09:00 09:00 - 10:00 10:00 - 11:00 11:00 - 12:00 12:00 - 13:00 13:00 - 14:00 14:00 - 15:00 15:00 - 16:00 16:00 - 17:00 17:00 - 18:00 18:00 - 19:00 19:00 - 20:00 20:00 - 21:00 21:00 - 22:00 22:00 - 23:00	08:00 - 09:00 09:00 - 10:00 10:00 - 11:00 11:00 - 12:00 12:00 - 13:00 13:00 - 14:00 14:00 - 15:00 15:00 - 16:00 16:00 - 17:00 17:00 - 18:00 18:00 - 19:00 19:00 - 20:00 20:00 - 21:00 21:00 - 22:00 22:00 - 23:00	08:00 - 09:00 09:00 - 10:00 10:00 - 11:00 11:00 - 12:00 12:00 - 13:00 13:00 - 14:00 14:00 - 15:00 15:00 - 16:00 16:00 - 17:00 17:00 - 18:00 18:00 - 19:00 19:00 - 20:00 20:00 - 21:00 21:00 - 22:00 22:00 - 23:00	08:00 - 09:00 09:00 - 10:00 10:00 - 11:00 11:00 - 12:00 12:00 - 13:00 13:00 - 14:00 14:00 - 15:00 15:00 - 16:00 16:00 - 17:00 17:00 - 18:00 18:00 - 19:00 19:00 - 20:00 20:00 - 21:00 21:00 - 22:00 22:00 - 23:00	08:00 - 09:00 09:00 - 10:00 10:00 - 11:00 11:00 - 12:00 12:00 - 13:00 13:00 - 14:00 14:00 - 15:00 15:00 - 16:00 16:00 - 17:00 17:00 - 18:00 18:00 - 19:00 19:00 - 20:00 20:00 - 21:00 21:00 - 22:00 22:00 - 23:00

Friday

The Friday is one of the more relaxed days of the week. A lot of students are going home, so it's quiet at CH. I've time to work on my biannual exploitation. In the lunch I have a conversation with a treasurer off one of the many commissions CH has. We talk about the financial status of his commission and whether or not he thinks it goes well. In the evening it's time to train off all those beers of last week, so I play some futsal on the sports centre.

So now you know what a normal week of my life as a board member looks like. The only problem is that there don't exist any normal weeks, because every week is special. Every week there are things that I just can't put in the scheme of a normal week. If you want to know more about all the exciting and thrilling activities that cross my path every week, you know where to find me on a 'normal' Wednesday late noon. 



Misverstanden

Prof.dr. C. Witteveen

Leest u vooral door. U zult merken dat ik maar weinig nodig heb om m'n eigen stokpaardje te berijden. En dat stokpaardje is algoritmie.

Nu gaat het me deze keer niet eens zozeer om de inhoud, als wel om de misverstanden rond dat onderwerp. Zo is bij een gedeelte van mijn kennissenkring de eerste associatie bij "algoritmie" die met een speciale vorm van ritmiek of gymnastiek. "Algo" wordt hooguit beschouwd als een poging zo'n fitnessmethode een wat exotisch tintje te geven. Ik leg dan geduldig uit dat ze er nog niet eens zo ver naast zitten. Hoogstens zou je kunnen zeggen dat algoritmie meer betrekking heeft op het bereiken van een mentale vorm van fitness. Verder ga ik meestal niet, want de vragen die ik dan krijg over doping in de algoritmie ga ik natuurlijk niet beantwoorden.

Nu zou je mogen verwachten dat een deskundig publiek beter geïnformeerd is. Maar, nee hoor. Zo had ik laatst een borrel waar ook een aantal IT-ers bij aanwezig was. Toen ik ook daar moest uitleggen waar ik me mee bezig houd, had ik de grootst mogelijke moeite om duidelijk te maken dat algoritmie en zoiets als programmeren of programmeertaalkunde totaal verschillende takken van sport zijn. Laat me het dan nog één keer uitleggen: in een programmeer-

taal leg je vast hoe je je op een nette manier verstaanbaar kunt maken in het computerese: een taal die een computer verstaat. Algoritmie gaat echter over slimme ideeën voor de oplossing van problemen die zo geformuleerd kunnen worden dat ze in een computertaal (maakt niet uit welke) zijn vast te leggen. Ok, ok, als dit te hoogdravend klinkt: kijk, een taal is een verzameling van spelregels waaraan je je dient te houden om je verstaanbaar te maken in b.v. een gesprek. In zo'n taal kun je benullige en onbenullige ideeën spuien, als je je maar aan de regels van de taal houdt. Dan hoor je een computer niet klagen. Algoritmie daarentegen probeert een scheiding aan te brengen tussen onbenullige en benullige ideeën en je hoort ons daar altijd over klagen.

Goed. Stel dat je dit soort misverstanden uit de weg hebt geruimd, dan kun je met een tweede categorie misverstanden geconfronteerd worden. Voorbeeld: laatst had ik een interessant gesprek met een neef die organisatiekunde gaat studeren, een heel respectabel vak zoals iedereen weet. Hij begreep zowaar na enige uitleg waar het in de algoritmie om draait. En vervolgens wist hij me te vertellen dat ook in dat vak algoritmie aan bod komt in de zin zoals ik hierboven het vak heb beschreven. Zo vertelde hij me dat een befaamde slogan: "organisatiekunde is de kunst van het slim organiseren" eigenlijk betekent dat men daar algoritmen (= slimme methoden) ontwikkelt die veel verder gaan dan waar ik mee bezig ben. Maar toen ik hem vroeg of de slimheid ook zo groot was dat je de uitvoering aan een computer kon overlaten, vond hij dat een belachelijk idee: het gaat om mensen die deze methoden moeten uitvoeren.

Ik vertelde hem dus dat ik mijn algoritmie een veel spannender vak vind, omdat je gedwongen wordt je slimme idee zo vorm te geven dat het geen enkele slimheid vereist om dat idee uit te voeren. Dan weet je ook zeker dat het precies zo uitgevoerd wordt als jij bedoeld hebt. Dat betekent dat je -in tegenstelling tot zijn kunde- geen creativiteit nodig hebt bij de uitvoering van zo'n idee. Een slim algoritme zegt dus alles over de bedenker ervan en niet iets over het algoritme zelf of de uitvoerder ervan.

Het kan echter nog erger. In iedere familie heb je ook wel een -loog. In de mijne in ieder geval wel, zelfs meer dan één. Je kunt ze vaak herkennen aan hun favoriete uitspraak: "Alles is gedrag", met als onuitgesproken vervolg: "dus mijn vak is het belangrijkste". Nu is zo'n uitspraak natuurlijk waar, net zo waar als wanneer je naar de sterrenhemel kijkt en opmerkt dat alle sterren even ver van ons af lijken te staan. Ook zo'n opmerking waar je de Nobelprijs niet voor zult krijgen. Maar belangrijker dan zo'n uitspraak "alles is gedrag" is natuurlijk de vraag hoe dat gedrag dan gegenereerd wordt: is gedrag als de uitkomst van een computationeel proces te beschrijven? Kunnen we daar slimme algoritmen voor maken? Welke beperkingen hebben dat soort algoritmen? Met andere woorden: -loog, zonder nou te beweren dat alles een algoritme is, zou jij niet eens bij ons langs moeten komen in plaats van andersom?

Ziezo, u heeft m'n raad opgevolgd en u bent helemaal aan het eind gekomen. Ik ben tevreden. Ik hoop dat u een paar misverstanden of illusies armer bent. Ik kan weer aan het werk, liefst zonder illusies. 



FSR

Marieke van der Tuin

Afgelopen tijd is de Facultaire Studentenraad hard aan de slag gegaan met het 10-puntenplan (zie het vorige MaCHazine). Om de week hebben we een interne vergadering. Daarnaast vergaderen we eens per maand met de decaan en opleidingsdirecteuren van EWI. Ook vindt eens per maand een facultair overleg plaats met de studentenraad en afgevaardigden van alle FSR's van de TU Delft. Kortom: we zitten niet stil!

Kluisjes

Zo hebben we ons beziggehouden met de kluisjes op EWI. Een veel gehoord probleem is dat de kluisjes altijd vol zitten, terwijl ze vaak helemaal niet actief gebruikt worden. De FSR heeft er daarom voor gezorgd dat er binnenkort een extra blok met kluisjes bij komt. Om het probleem niet alleen tijdelijk op te lossen, hebben we ook afgesproken dat de kluisjes elke 14 dagen zullen worden geleegd vanaf het moment dat het nieuwe kluisjesblok er is. Zo zorgen we er hopelijk voor dat de kluisjes weer zullen worden gebruikt voor hetgeen ze bedoeld zijn: tijdelijke opslag van eenieders spullen.

Nieuw curriculum

Verder heeft de FSR zich natuurlijk ook beziggehouden met de onderwijsvernieuwingen. In het collegejaar 2013-2014 zal het nieuwe curriculum voor de bachelors Technische Informatica en Technische Wiskunde worden ingevoerd. Alle jaarlagen zullen tegelijkertijd overstappen op dit nieuwe curriculum. Niet alleen de vakken en het aantal punten per vak zal veranderen: ook de academische jaarindeling gaat op de schop. Voortaan zal een kwartaal bestaan uit 4 weken college, 1 week colleges of tentamens gevolgd door nogmaals 4 weken college en een week tentamens. De 'witte week' verdwijnt hierbij. Zo'n nieuwe structuur vraagt natuurlijk ook om een heel andere aanpak met het toetsen van de stof. Daarom heeft de FSR hier een voorstel voor geschreven en deze aangeboden aan de opleidingsdirectie. Het uitgangspunt van dit voorstel is ervoor te zorgen dat studenten gemotiveerd worden om de stof goed bij te houden zodat vakken sneller kunnen worden voltooid en het aantal herkansingen beperkt blijft.

BSA

Ook het verhoogde BSA dat dit collegejaar is ingevoerd houdt de FSR bezig. Alle eerstejaars moeten dit jaar maar liefst 45 EC behalen om verder te mogen gaan met hun studie. De cijfers naar aanleiding van de tentamens van het 1e kwartaal hebben we goed bekeken. Daaruit bleek echter dat we hier nog nauwelijks iets mee konden, gezien de grote hoeveelheid semestervakken. Nadat alle resultaten van het 2e kwartaal ook binnen zijn, zullen we bekijken in hoeverre de studenten op dit moment goed op weg zijn om hun BSA te halen. Mocht dit erg tegenvallen, dan kunnen we proberen om een vak uit te zonderen van het BSA en eventueel een extra herkansing afdwingen.

Promotie

De FSR heeft ook geprobeerd om zichzelf kenbaar te maken aan de rest van EWI. Zo plaatsen we op onze Facebook-pagina (<http://www.facebook.com/EWI.FSR>) regelmatig waar wij mee bezig zijn of wat we hebben bereikt. Daarnaast hebben we op 20 december een informatiebijeenkomst georganiseerd waar alle studenten hun stem konden laten horen en ons konden vertellen wat hen bezig houdt. Tijdens dit collegejaar zullen we nog een dergelijke informatiebijeenkomst organiseren.

Wisseling van de wacht

Binnenkort zal de FSR voor volgend jaar weer worden gekozen. Als je ook graag bij de FSR zou willen, meld je dan snel via fsr@ch.tudelft.nl of spreek iemand van de FSR aan voor meer informatie. Op 22 en 23 meikan iedereen stemmen op de mensen die zich hebben aangeboden voor de FSR. Ook kun je dan weer je stem uitbrengen op de verschillende partijen voor de studentenraad. Laat van je horen en stem zodat ook jouw mening volgend collegejaar vertegenwoordigd wordt! 🗳️



STIP

Anne Viruly

Wie zijn jullie?

Wij zijn STIP. Studenten Techniek In Politiek, een progressieve politieke partij in de Delftse politiek. De naam zegt het al: de partij wordt gerund door studenten. Dat zie je ook goed terug: in de gemeenteraad hebben wij een jonge, moderne kijk op alles wat er in de stad speelt.

En wat doen jullie?

Wij vertegenwoordigen j  uw stem in de Gemeenteraad. STIP is fulltime te vinden in het stadhuis op de Markt. Je bent daar welkom voor al je vragen, problemen en idee  n rond de stad. Via de politiek gaan wij dan voor je lopen om het probleem op te lossen, of je goede idee werkelijkheid te maken. Op het moment zijn we daarom bezig om meer fietsenstallingen te plaatsen voor verschillende soci  teiten. Ook zetten we ons hard in om de kamernood terug te dringen en zullen er tussen 2010 en 2014 meer dan 2000 studentenwoningen worden bijgebouwd.

Hoe doen jullie dat?

STIP bestuurt echt mee in de stad. Bij de vorige verkiezingen kregen wij maar liefst 3 van de 37 zetels in de raad. Na de verkiezingen hebben we samen met D66, de PvdA, GroenLinks en het CDA een coalitie gevormd van vijf partijen, en leveren we ook een wethouder. Met dit aantal zetels en onze positie in het college van Burgemeesters en Wethouders, zijn wij in Delft in de positie dingen te veranderen en te verbeteren.

Wat hebben jullie dan zoal bereikt?

Zal ik gewoon een paar voorbeelden noemen? Mede door de inzet van STIP is YES!Delft opgericht, zodat studenten hulp kunnen krijgen bij het opstarten van een bedrijf. Ook hebben we ervoor gezorgd dat er een tram komt die je van het station naar de TU brengt, en dat je overal in de Binnenstad mag fietsen. Door STIP mogen caf  s vaker lang open blijven, komen er meer studentenwoningen en zijn er steeds meer oplaadplekken voor elektrische auto's.

Politiek, doet iedereen dat fulltime?

Nee, alleen wij. STIP gelooft gewoon niet in de zogenaamde Ivoren Toren. Wij nemen daarom de tijd om echt met mensen te praten. En doordat wij als studenten even fulltime kunnen gaan, kan dat ook. Maar het heeft ook andere voordelen. Bij nieuwe plannen van de gemeente kunnen wij eens rustig bellen

met een andere gemeentes die dat al doen - "is dat een succes bij jullie?" - en kunnen we er veel over uitzoeken. Dit zorgt ervoor dat het alom bekend is dat STIP altijd goed voorbereid is.

En hoe werkt de gemeenteraad eigenlijk?

Elke laatste donderdagavond van de maand vergadert de Gemeenteraad op het Stadhuis. Onder leiding van de burgemeester vergaderen daar 37 politici, verdeeld over elf politieke fracties. Als de raad echter in die avond alle voorstellen, nota's en stukken van die maand zou willen behandelen, zou de vergadering de hele dag duren. Daarom worden alle stukken eerst besproken in vier vakcommissies.

Als daar blijkt dat men het niet eens kan worden, wordt er verder gedebatteerd en eventueel gestemd in de Raad.

Zoeken jullie nog mensen?

Altijd! De verkiezingen van 2014 komen eraan, waar we een nieuwe kieslijst voor gaan samenstellen. Dus, heb jij politieke interesse, idee  n voor de stad, of lijkt het je wat om een politieke campagne te organiseren? In een artikel kunnen we lang niet uitleggen wat we allemaal doen. Kom dus eens een kopje koffie drinken op het stadhuis! Bel ons, kijk op www.stipdelft.nl, of stuur een mailtje naar stip@delft.nl.

En vergeet vooral niet te stemmen, in 2014!

STIP-fractie, vlnr: Bert Vogel, Annemieke Pannekoek, Gennadij Kreukniet, Maarten Buskermolen, Ruben Schippers en Anne Viruly. Rechtsboven wethouder Pieter Guldemon.



ORAS

Eva Nieuwenhuis

Ik ben Eva Nieuwenhuis, vierdejaars IO-student, en dit jaar zit ik fulltime namens ORAS in de Studentenraad. Misschien weet jij al precies wat ORAS en de studentenraad doen, misschien ken je ons alleen van de campagnes in mei of misschien ken je ons wel helemaal niet...

De Centrale Studentenraad vertegenwoordigt jou en alle andere 17.000 studenten aan de TU Delft. Samen met 9 anderen zet ik me dit jaar fulltime in om jouw stem zo goed mogelijk te laten horen. De Studentenraad is een wettelijk orgaan en heeft ook een heleboel rechten. Op eigenlijk alles wat met studenten te maken heeft, heeft de Studentenraad adviesrecht. Daarnaast heeft ze op een aantal zaken binnen de TU instemmingsrecht, zoals bijvoorbeeld het studentenstatuut.

Om dingen te bereiken moet je natuurlijk jouw ideeën en meningen communiceren naar de TU. Dit doen we door onder andere gesprekken te voeren met allerlei verschillende medewerkers binnen de TU; van medewerkers bij Marketing en Communicatie, tot beleidsmedewerkers bij Onderwijs en Studentenzaken tot alle drie de leden van het College van Bestuur van de universiteit.

Tot zover de studentenraad in het algemeen, maar wat is ORAS dan? ORAS is een van de twee partijen in de Studentenraad en elk jaar in mei worden er verkiezingen gehouden om de tien zetels te verdelen. Tijdens de laatste verkiezingen heeft ORAS zes van de tien zetels gehaald. ORAS zet zich in voor onderwijs van hoge kwaliteit en wij denken dat goede ondersteunende faciliteiten, zoals mooie werkplekken en een UB, hierbij onmisbaar zijn. Daarnaast

vindt ORAS dat een student zich moet kunnen ontplooiën tijdens zijn studie, bijvoorbeeld door het organiseren van of het meegaan op studiereis, maar ook door een project als Nuna of een commissie op je studentenvereniging.

Op dit moment zijn we bezig met zeer uiteenlopende zaken. Zo heeft ORAS contact met Studentenraden uit Leiden en Rotterdam om te praten over het samenwerkingsverband met Delft. Daarnaast heeft ORAS een samenwerking afgesloten met andere Studentenraden om op te komen voor de belangen van bèta studenten. Verder is ORAS betrokken bij de ontwikkeling van nieuwe complexen van het Sport- en Cultuurcentrum en denken we mee over RAS maanden en alle problemen en ontwikkelingen op ICT gebied. Ook is ORAS bezig met de nieuwe curricula, de evaluatie van het BSA, nieuwe selectie procedures voor aspirant studenten aan de TU en nog veel meer...

Natuurlijk willen we ook graag weten wat jij vindt. Wellicht ben je ons de afgelopen maand tegengekomen en heb je de enquête over docentkwaliteit ingevuld? Wil je op de hoogte blijven van wat we allemaal doen? Like ons dan op Facebook!

goed moment dus om een kop lekkere, warme chocolademelk te pakken, de verwarming een tandje hoger te zetten en er weer helemaal voor te gaan om het onderwijs op de TU nog mooier te maken dan het al is.

Want waar houdt Lijst Bèta zich in de Studentenraad momenteel mee bezig? De gehele Studentenraad is momenteel druk bezig om samen met de FSR'en de vorming van de nieuwe curricula in goede banen te leiden en om samen voor een goede overgangsregeling te zorgen. Daarnaast heeft de Studentenraad het bijna voor elkaar dat alle vakken die in het laatste half jaar van de Bachelor worden gegeven ook herkanst kunnen worden in het eerste half jaar, om zo de eventuele vertraging die studenten kunnen oplopen door de Harde Knip met een half jaar te reduceren.

Lijst Bèta zet zich in om het onderwijs voor studenten uitdagender en studeerbaarder te maken. Door de snelle groei van de digitale wereld komen er steeds meer mogelijkheden om het onderwijs te verbeteren, Lijst Bèta vindt dat deze nieuwe mogelijkheden met beide handen aangepakt moeten worden.

Lijst Bèta

Tristan Steegman

Met kerst en oud en nieuw achter de rug, zijn wij als Lijst Bèta met veel energie begonnen aan het nieuwe jaar. De eerste sneeuw van het jaar is gevallen en het Mekelpark is bedekt met een mooie witte deken. Een



Zo is men op de TU bezig, op initiatief van Lijst Bèta, met het ontwikkelen van interactieve, slimme oefenopgaven. Hierbij kun je denken aan hints op drie verschillende niveaus, feedback over gemaakte fouten en afspeellijsten met modules waardoor je zelf kan kiezen welke typen opdrachten je nog wilt oefenen. Verder komt er een systeem waarmee je snel je niveau binnen een vak kan bepalen door het maken van één of twee opgaven en zo naar makkelijkere of moeilijkere modules binnen het vak verwezen wordt. Deze opgaven zijn vanaf de derde periode beschikbaar voor het eerste vak, structural analysis

op LR, in een pilot vorm. Hierna zullen vergelijkbare opgaven voor andere wiskunde vakken gemaakt worden. Het is de bedoeling dat uiteindelijk voor elk exact vak aan de TU deze opgaven beschikbaar zijn. Wij wensen iedereen nog een fantastisch mooi en leerzaam 2013! 

VSSD

Fokke de Groot

De VSSD zet zich elke dag hard in voor jou als student. Op landelijk gebied zetten wij ons hard in om de regering te stoppen. Wij lobbyen heel hard tegen het leenstelsel en als het er toch komt pleiten wij voor een jaar uitzondering voor meerjarige masters.

Daarnaast zijn wij voor het behoud van het OV en zijn van mening dat het maximum van het OV weer naar 8 jaar moet. Een paar weken terug is er een wetvoorstel naar de Tweede Kamer gegaan waarin helaas, bijna onvermijdelijk, toch het leenstelsel is aangekondigd. Ons harde werk is echter niet helemaal voor niets geweest, want de Minister heeft daar ook in aangegeven te willen onderzoeken of er een uitzondering kan komen voor meerjarige masters. Dit betekent niet dat alle master stufi behouden zal worden maar dat er in het geval van de Delftse student een uitzondering komt van 1 jaar. De Minister gaat deze mogelijkheid echter eerst nog onderzoeken met haar collega Minister van Economische Zaken. Wat betreft het verdwijnen van de OV jaarkaart is

er ook een beetje positief nieuws, want ook hier lijkt een compromis uit te komen. Zoals we het nu kunnen inschatten gaat de OV jaarkaart vervangen worden door een trajectkaart. Hierdoor kan je waarschijnlijk alleen nog maar gratis reizen tussen je ouders en Delft en krijg je korting op andere trajecten.

Genoeg over de Landelijke politieke spelletjes, want de VSSD maakt zich op alle vlakken sterk voor de Delftse student. Dat houdt dus ook in dat wij ons bezighouden met veranderingen in het onderwijs op de TU Delft. Zo is de TU Delft bezig om ervoor te zorgen dat studenten sneller afstuderen door allerlei maatregelen te treffen die over het algemeen nadelig zijn voor de student. Denk aan de verhoging van het BSA, blokonderwijs en compensatoir toetsen. De verhoging van het BSA naar 45 lijkt te hoog gegrepen en onrealistisch voor de gemiddelde student. Om uitval te bestrijden heeft de TU hier bovenop blokonderwijs en compensatoir toetsen bedacht. Compensatoir toetsen lijkt voordelig omdat je dan tentamens mag compenseren met andere tentamens, maar leidt al snel tot devaluering van je diploma. Blokonderwijs vormt geen gevaar voor de devaluering van je diploma maar brengt weer de nadelen met zich mee dat je 2x zoveel tentamens hebt en langere dagen op de TU doorbrengt, waardoor het lastig wordt om nog een sociaal leven te hebben. Kortom genoeg om het CvB over aan de tand te voelen en te overtuigen van de nadelen van hun maatregelen.

Wil jij meedenken en helpen bij de mening van de Delftse student formuleren: stuur dan een mail naar bestuur@vssd.nl 



TU Delft nieuws

Max de Groot

De TU Delft is de grootste en oudste Nederlandse publieke technische universiteit en is opgericht door Koning Willem II op 8 januari 1842. Hierdoor heeft de TU Delft onlangs zijn 171ste Dies Natalis gevierd met als thema 'gezondheid'. Maar wat gebeurt er momenteel verder op de TU Delft? In dit artikel zetten we de belangrijkste gebeurtenissen van de afgelopen tijd op een rijtje.

Massive Open Online Course

Steeds meer gerenommeerde universiteiten wereldwijd publiceren hun beste cursussen als 'Massive Open Online Course' (MOOC). Er bestaan inmiddels meerdere platforms waarop MOOCs gepubliceerd worden, zoals Udacity, edX en Coursera. Ook de TU Delft zal dit jaar met MOOCs starten en maakt binnenkort bekend op welk platform die gepubliceerd zullen worden.

MIT-hoogleraar en TU Delft alumnus Walter Lewin voegt zijn 'Electricity and Magnetism' (MIT 8.02x) cursus toe aan edX. Dit maakte EdX op dinsdag 22 januari 2013 bekend. EdX is een platform voor online onderwijs, waarmee onder andere MIT en Harvard sinds 2012 cursussen (MOOCs) toegankelijk willen maken voor iedereen ter wereld met toegang tot internet. Deelname is voor iedereen toegankelijk, ook zonder vooropleiding of ingangstoets.

Impuls voor Delftse medische technologie

Financier van gezondheidsonderzoek ZonMW heeft projectfinanciering toegekend aan drie door de TU Delft geleide centra van het Innovative Medical Devices Initiative (IMDI) voor de ontwikkeling van nieuwe en verbeterde medische technologie. Het gaat om projecten op het gebied van minimaal invasieve technologie, neurorevalidatie, en beeldvormingstechnologie voor cardiovasculaire en neurodegeneratieve aandoeningen. Het totale bedrag van het zogenoemde IMDI-programma 2012-2013, met financiering vanuit de topsector LSH en uitgevoerd door ZonMW, bedraagt 5 miljoen euro. De centra ontvangen samen 1,9 miljoen euro. Deze toekenning stelt de onderzoekers in staat om deze technologieën via publiek-private samenwerking versneld te ontwikkelen en op de markt te brengen.

Minimaal invasieve technieken maken een diagnose en behandeling minder ingrijpend en veroorzaken zo een revolutie in de medische zorg. Bij deze technieken zijn er echter nog heel wat uitdagingen: op veel plekken in het lichaam kan men niet komen en er zijn zeer beperkte mogelijkheden voor het tegelijkertijd uitvoeren van diagnose en behandeling. Bovendien is het ontwerpen van prototypes om uit te testen in de patiënt nu nog erg lastig en te tijdrovend. In het NIMIT Development Center worden met behulp van de impulsfinanciering prototypes gemaakt van drie innovatieve instrumenten: een wrijvingsloze grijper, een stuurbare kniptang en 3D beeldvorming voor fetoscopie.

Delftse looprobots gaan nu ook rennen

De looprobots van de TU Delft zijn nu ook voorzichtig toe aan rennen. Doel is vooral om de menselijke renbeweging beter te leren begrijpen. Onderzoeker Daniël Karssen bouwde hiervoor de robot Phides. Karssen promoveert op vrijdag 18 januari op dit onderwerp.

Het onderzoek van Daniël Karssen, dat mede werd gefinancierd door Technologiestichting STW, heeft vooral als doel om de menselijke renbeweging beter te begrijpen. 'Het begrijpen van de menselijke renbeweging is noodzakelijk voor de ontwikkeling van hulpmiddelen die gehandicapte mensen laten rennen, zoals hardlooprotheses, en voor hulpmiddelen die gezonde mensen beter laten rennen', zegt Karssen. 'Deze hulpmiddelen worden nu al wel ontwikkeld, maar het ontbreekt nog aan fundamenteel inzicht in de renbeweging. Een grote onbekende factor vormen verstoringen, zoals een onverwachte duw of hoogteverschillen in de ondergrond.' Zie de robot rennen op: <http://wisv.ch/runningrobot>

De tramlijn in het Mekelpark komt eraan...

Maandag 14 januari 2013 zijn de werkzaamheden voor tramlijn 19 in het Mekelpark hervat. In opdracht van Stadsgewest Haaglanden voeren aannemers op de TU Delft campus werkzaamheden uit voor de aanleg van Tramlijn 19. Van maandag 14 januari tot en met vrijdag 8 februari werd gewerkt in het gebied Aula/IO tot aan CiTG/EWI. Hierna wordt het werkgebied uitgebreid tot aan de Balthasar van der Polweg.

De Delftse Bedrijvendagen

De Delftse Bedrijvendagen is de grootste technische carrièrebeurs van Nederland en vindt elk jaar plaats in de Aula van de TU Delft. Deze beurs is speciaal gericht op de Delftse student en biedt een grote variëteit aan bedrijven en instellingen. Op 12, 13 en 14 februari waren de Sollicitatiedagen van topbedrijven; op 19, 20 en 21 februari waren de presentatiedagen en kon je kennis maken met meer dan 100 (inter)nationale bedrijven; vanaf 18 maart tot en met 5 april zijn de inhouse dagen om meer te komen weten over een bedrijf; vanaf 24 april t/m 8 mei zijn de gesprekkendagen, zodat je één-op-één gesprekken hebt met een bedrijf.

Het sociaal-leenstelsel

Het leenstelsel zal per collegejaar 2014/2015 worden ingevoerd. Dat betekent dat als je in dat jaar of later begint met je bachelor ofwel je master, je geen studiefinanciering meer zult krijgen en alleen nog maar kunt lenen. Dit is vooral van belang voor studenten uit 2009/2010/2011, die nu moeten proberen om zich nog tijdens collegejaar 2013/2014 in te schrijven voor een master zodat zij niet getroffen zullen worden door het leenstelsel.

De mogelijkheid tot het behoud van één jaar basisbeurs voor studenten met een meerjarige master is iets waar de VSSD en de Studentenraad zich voor blijven inzetten. 

Referenties

- [1] <http://www.tudelft.nl/nl/actueel/laatste-nieuws/>
- [2] <https://ddb.tudelft.nl>



Docent van het Jaar

Joost de Groot

Ik ben verkozen tot “TU Delft docent van het jaar”, uiteraard een enorme eer! Maar waarom is mij die eer ten deel gevallen? Het echte antwoord weet ik niet, maar mijn visie hierop en mijn ervaringen hiermee probeer ik op deze bladzijde te verwoorden.

Er is mij gevraagd over het onderwerp uit de titel iets te schrijven. Ondanks het feit dat docenten graag aan het woord zijn is dit geen eenvoudige vraag. Het is immers iets totaal anders dan een lesje wiskunde. Dat lesje staat namelijk al in de boeken, wij hoeven slechts na te praten. Ik heb het idee dat ik dat ook doe. Vandaar dat ik me serieus af vraag waarom ik dan toch die titel gekregen heb. Een analyse:

Wat is les geven voor mij? Natuurlijk begin je dan met de open deur dat het hoofddoel is om een hoeveelheid leerstof in “hapklare brokjes” over te brengen, waarbij ik als docent degene ben die elke vraag over de stof zou moeten kunnen beantwoorden en degene die in staat is om geschikte opgaven te selecteren waarmee de kennis van de theorie niet alleen getoetst, maar ook uitgebreid kan worden.

Maar voor mij is er meer: ik wil studenten laten zien hoe mooi wiskunde is! Ikzelf ben er mateloos enthousiast over (zelfs als ik een vak, zoals lineaire algebra, al vele, vele, vele malen gegeven heb). Dat enthousiasme hoop ik over te brengen op de student. Ik ben ervan overtuigd dat dat, zeker bij eerstejaars, niet lukt als je ingewikkelde bewijzen van (vaak hele mooie) stellingen op het bord zet, zonder dat de student het belang en de voorwaarden van de stelling overziet, maar wel als je laat zien hoe de theorie in elkaar zit, mogelijk aan de hand van één of meer voorbeelden. Mijn ideaal: naast een collegeserie een verdiepend college waarbij ingegaan wordt op de vaak prachtige bewijzen van de hoofdstellingen, maar pas nadat deze al toegepast zijn op opgaven. Helaas zal dit in de nabije toekomst, maar waarschijnlijk ook op de langere termijn, niet mogelijk zijn.

Verder zie ik een college ook als een sociaal gebeuren, een plek waar studenten elkaar en hun docent ontmoeten. Er moet sfeer in zitten. Ik vind dat je als docent tussen de studenten moet staan, er moet een vertrouwensband ontstaan. Vandaar dat ik altijd probeer op tijd aanwezig te zijn, tijdens de pauze persoonlijke vragen wil beantwoorden (met een kopje koffie, wat er wel eens bij inschiet) en aan het eind van het college niet direct weg ga (tenzij ik naar een volgende les moet).

Ook is mijn devies: wij zijn er voor de student en niet andersom!! De bedoeling is dat de student het vak gaat begrijpen. En hoewel het wel eens moeilijk is, zullen wij toch moeten afdalen naar het niveau van de student, ook als we het vak kennen als onze broekzak.

En dit alles lijkt te lukken, ik denk namelijk dat ik daarom die titel gekregen heb. Ik word vaak aangesproken over mijn enthousiasme en heldere uitleg en hoewel ik, zoals eerder gezegd, denk dat ik het boek na praat, beweren studenten dat dat niet waar is. Dat zal dan wel...

Nu iets over de verkiezing: het is overweldigend wat er allemaal gebeurd is. Het begon met de verkiezing tot docent van het jaar 2011-2012 tijdens een ADSL (Algemene Docenten en Studenten Lounge). Vier tot zes keer is gevraagd of ik ook kon komen en telkens weer moest ik uitleggen dat ik niet kon. Helaas was ik zelf dus niet bij de ADSL aanwezig. 's Avonds feliciteerde Tomas Klos mij via facebook, en, naïef als ik ben, had ik niet door wat hij bedoelde. Een paar weken later ontving ik de mij welbekende wisselbeker waarin de namen van de docenten van het jaar gegraveerd staan. Hiermee was het bewijs definitief geleverd, qed.

En daarna? In september kwam de directeur onderwijs vertellen dat de opleidingsdirectie van EWI mij heeft uitgeroepen tot docent van het jaar van EWI. Een hele eer, maar ook een opdracht voor de commissaris onderwijs van CH. Max de Groot moest zorgen voor een hele lijst handtekeningen als bewijs dat studenten achter mijn verkiezing staan, hij moest een betoog schrijven waarom ik de titel docent van het jaar van de TU verdien, hij moest zorgen voor een promotiefilm over mij (en dat heeft hij gedelegeerd naar Paul Brussee en Arjen Wiersma; een geweldige keus, want zelfs ik vond het een werkelijk magnifiek filmpje. Paul en Arjen, nogmaals bedankt!!) en hij gaf mij de opdracht een curriculum vitae en een overzicht van mijn onderwijs te maken. Dit alles om de jury die de beste docent verkiest te overtuigen. En op 29 november bleek dat het is gelukt! Max, ook jij nogmaals bedankt, en natuurlijk ook al degenen die een woordje geschreven of gezegd hebben, te veel studenten om op te noemen!

Op het moment van de bekendmaking gingen er twee gedachten door me heen. Natuurlijk zoets als “yes”, en “wow”, maar ook “oef, nu moet ik speechen, in het engels...”, en ik heb niets voorbereid.” We (de acht genomineerden) hadden van tevoren te horen gekregen dat de winnaar iets moest gaan zeggen. Opnieuw was ik in de maling genomen door mijn naïviteit: ik had niets voorbereid omdat ik ervan overtuigd was dat ik het niet zou worden. Daarvoor waren genoeg redenen: ik houd van krijtborden, ik ben een ouderwetse docent, ik ben geen hoogleraar en ik ben geen vrouw.

Hoewel ik het eigenlijk niet wil zeggen omdat ik een hekel heb aan kapsones, zeg ik het nu toch: de jury vertelde me dat het volstrekt duidelijk was wie het moest worden! Unaniem zijn ze tot hun keuze gekomen (zeiden ze). Dat gaf me uiteraard een goed gevoel. Mijn dank aan de studenten voor al hun inzet waardoor de jury overtuigd werd, is daarbij groot.

En het meest frappante? Dat letterlijk de eerste die je feliciteert op het moment dat je de zaal uit bent een volstrekt onbekend iemand is: “ik ben journaliste van de delta en wil je zo snel mogelijk interviewen, wanneer kun je? Uiterlijk maandag! Ik wil nu meteen een tijdstip afspreken.” (de uitreiking was op donderdag).

Ik zal jullie verder niet lastig vallen met alle andere zaken die ook nog gebeurd zijn, het is al erg genoeg dat ik daarmee lastig gevallen ben ;-)



Wintersport '13

Kasper Kooijman

Op 1 februari was het eindelijk zo ver, na de laatste tentamens stonden 54 CH-ers te wachten op de bus die om half 7 richting de Franse Alpen zou vertrekken. Met de gebruikelijke vertraging van Biztravel werd het acht uur, maar desalniettemin was de vakantie nu echt begonnen.

Het gebied *Les Sybelles* was voor zowel beginnend als ervaren wintersporter een prima skigebied. Met 300 pistekilometers en genoeg variatie aan blauwe, rode en zwarte pistes, konden de ervaren skiërs en snowboarders hard werken op de zwarte pistes en genieten van de rode, terwijl de beginners voldoende blauwe pistes hadden om op te oefenen.

Behalve heel veel skiën en snowboarden, is er natuurlijk ook ontzettend veel geëist. Nadat iedere avond in verschillende huisjes alle post-it briefjes van de muren waren gedronken, was het meestal tijd om een bezoekje te brengen aan de Viebar in het dorp. In deze Viebar is op dinsdagavond het WiFi WinterWonderFeest gehouden, waar buiten een enorme lading aan CH-ers ook een hele hoop Belgen, een paar Amerikanen en een zooi andere Delftenaren op af zijn gekomen. Na een lange avond met veel drank, heel veel drama, vallende leeuwen en een geweldig leuk feest, werd de volgende dag voor een enorme sneeuwdump gezorgd, wat van de woensdag voor veel CH-ers een prima uitbrakdagje maakte.

Na deze 'brakke woensdag' moest van de donderdag natuurlijk optimaal gebruik worden gemaakt. Nadat er door vrijwel iedereen was genoten van de verse sneeuw van de dag ervoor, was iedereen die middag op de tafels van de Viebar te vinden tijdens de Après-ski waar het bier weer rijkelijk vloeide.

Inmiddels beseften veel mensen dat de week alweer bijna ten einde kwam. Op vrijdag werd nog een keertje uitgebreid genoten van het skigebied, maar op zaterdag moesten alle huisjes alweer vroeg schoon zijn. Sommigen zijn die dag nog lekker de piste opgegaan, terwijl anderen genoeg namen met het aangename zonnetje dat die dag scheen. Met een voldaan gevoel ging iedereen die avond uitgeput de bus in. We kunnen spreken van een geslaagde eerste WiFi Wintersport en we kijken nu al uit naar volgend jaar. De WiFi zal er in ieder geval weer bij zijn! 





CH Kerstdiner '12

Rebecca Jacobs

Op 20 december 2012 was het weer zover, het jaarlijkse kerstdiner van

W.I.S.V. 'Christiaan Huygens' kon beginnen!

Er waren ook dit jaar natuurlijk weer veel leden, ereleden en LvV'ers

aanwezig om onder het genot van goede spijsen en dranken bij te praten.

Entree

Wij maakten onze opwachting in restaurant De Kromme te Delft, waar wij vanaf 18.56 uur verwacht werden. Dit sfeervolle restaurant in de Kromstraat werd het toneel voor ons stijlvolle kerstmaal. De barre weersomstandigheden van deze avond zorgden ervoor dat rond een uur of zeven de eerste mensen rustig binnen kwamen, en hun vochtige jassen ophingen, om vervolgens ergens plaats te nemen en een beetje op te drogen. In totaal vormden alle gegadigden een zeer gemêleerd publiek, met aanwezigen van alle verschillende leeftijden en van verschillende opleidingen. De tweedejaars commissies waren opvallend goed vertegenwoordigd, aangezien twee van de drie commissies die geheel uit tweedejaars bestaan voltallig aanwezig waren.

Feestmaal

Nadat de eerste drankjes gedronken waren kwam het personeel langs om onze bestellingen op te nemen. De keuze van het voorgerecht bestond uit rundercarpaccio of tomatensoep. Het was onmogelijk een verkeerde beslissing te nemen, aangezien beide voorgerechten voortreffelijk smaakten. Als hoofdgerecht kon er wederom gekozen worden uit een aantal gerechten, namelijk: gebakken varkenshaas, zalm of een vegetarisch pastagerecht, allen geserveerd met salade en huisgemaakte frites. Nadat alle bestellingen waren opgenomen, hoefden we niet lang te wachten voordat we aan het voorgerecht konden beginnen.

Kerstverhaal

Nadat iedereen het voorgerecht op had en de borden waren afgehaald, was het weer tijd voor een belangrijke traditie die dit jaar een alternatieve draai kreeg: het kerstverhaal. Dit wordt ieder jaar geschreven en voorgedragen door de huidige voorzitter van W.I.S.V. 'Christiaan Huygens'. Helaas waren door ongelukkige omstandigheden zowel voorzitter Karens Grigorjancs als vice-voorzitter Joost Verkade niet aanwezig bij het kerstdiner. Gelukkig had Joost

ondanks zijn afwezigheid nog wel tijd gevonden om het verhaal te schrijven, en droeg de penningmeester Romke Rozendaal het verhaal met verve voor. Nadat Romke klaar was het zijn prachtige en ontroerende verhaal, werd het woord doorgegeven aan de aanwezige oud-besturen, een paar ereleden en zelfs een tweedejaars commissie. Iedereen werd uitgebreid bedankt voor zijn of haar aanwezigheid en inzet tijdens het afgelopen jaar.



Wijn, spijsen, koffie ... en meer wijn

Ondertussen werd het hoofdgerecht geserveerd, en nadat ieder die daar behoefte aan had - en enkelen die daar geen behoefte aan hadden - het woord hadden gehad en doorgegeven, kon er eindelijk aan het eten begonnen worden. Onder het genot van vele glazen wijn werd er gegeten en het afgelopen jaar besproken, maar ook gespeculeerd over het aankomende jaar.

Enige tijd nadat iedereen zijn of haar hoofdgerecht op had werden de borden weer opgehaald en kwam het personeel weer langs om de bestellingen voor het toetje op te nemen. Ditmaal de keuze uit een grand dessert, koffie of thee. Een tijdje later werd dit uitgeserveerd en onder het genot van wederom een glas wijn werd ook het dessert dankbaar ontvangen. Langzamerhand begonnen mensen te vertrekken. Een flink aantal van de gegadigden besloot nog naar cafe De Kurk te gaan om na te praten over dit gezellige kerstdiner. Al met al een zeer geslaagde avond met goed eten en leuk gezelschap. 



Association



Association



Alle vrijheid.

www.tamtam.nl/werkenbij

Tam Tam



A really fast GPU solver for real-time computation of interactive waves

Martijn de Jong

Although water visualization models from the film industry (e.g., Waterworld, Titanic) and modern ship simulators can look really fantastic, they often lack physical realism or are not suited for real-time simulation. For my Master's project I did research at MARIN on improving the speed of a new class of ship simulators [1] which are both physically correct and allow for real-time simulation.

In modern real-time ship simulators, the waves are deterministic (Fourier spectrum). To bring real-time ship simulators to the next level, a new interactive wave model is proposed by G. Klopman [2], called the Variational Boussinesq model (VBM), which allows ships and waves to really interact; i.e., ship motions are induced by the waves and the waves in their turn are affected by ships, moles, breakwaters, piers, etc. Moreover, the model can handle deep waters with varying depth (3D). Logically, the higher realism comes at a price: the new model is much more computational intensive. To be able to fulfill the requirements of real-time simulation (25 fps) a really fast solver is needed. The primary goal of my Master's project was to develop this solver.

Model description

The governing linearized VBM equations are given by [2]:

$$\frac{\partial \zeta}{\partial t} + \nabla \cdot (\zeta \mathbf{U} + h \nabla \varphi - h \mathcal{D} \nabla \psi) = 0, \quad (1a)$$

$$\frac{\partial \varphi}{\partial t} + \mathbf{U} \cdot \nabla \varphi + g \zeta = -P_s, \quad (1b)$$

$$\mathcal{M} \psi + \nabla \cdot (h \mathcal{D} \nabla \varphi - \mathcal{N} \nabla \psi) = 0, \quad (1c)$$

where Equation (1a) is the mass conservation equation which describes the evolution in time of the free surface (water level) $\zeta(x, y, t)$ and Equation (1b) is the Bernoulli equation which describes the evolution in time of the free surface velocity potential $\varphi(x, y, t)$. The third equation is an elliptic equation for the free surface vertical velocity $\psi(x, y, t)$. Furthermore, $\mathbf{U}$ is the horizontal flow velocity, h the water depth, g the gravitational acceleration, P_s the "footprint" of the ship (in $[m^2 s^{-2}]$), and $\mathcal{D}, \mathcal{M}, \mathcal{N}$ are model parameters. Because the vertical structure of water flows is often known, the velocity potential in the Bernoulli equation is expanded in cosine-hyperbolic shape functions. By doing so the 3D model in Cartesian coordinates (x, y, z) can be reduced to the above 2D model in (x, y) which yields a great reduction in computational effort.

The VBM equations are discretized with the finite volume method (FVM) on a Cartesian grid. Discretization leads to:

$$\frac{d\mathbf{q}}{dt} = L\mathbf{q} + \mathbf{f}, \quad (2)$$

$$S\vec{\psi} = \mathbf{b}. \quad (3)$$

Equation (2) is solved with the Leapfrog method, a second order explicit time integration method. System (3) is a linear system that has to be solved multiple times per second. The matrix S in (3) is real-valued, sparse (5-point, pentadiagonal), diagonally dominant, symmetric positive definite (SPD) and moderately large. For example, for a square domain of 10×10 km with a mesh spacing of 5 m, we have 4 million grid points and hence matrix S has size 4 million by 4 million.

A very efficient linear solver

For systems having a matrix S with properties as listed above, a preconditioned conjugate gradient (PCG) solver is very proficient [3], see Algorithm 1. Rather than solving system (3) we solve the left-preconditioned system

$$M^{-1}S\vec{\psi} = M^{-1}\mathbf{b}, \quad (4)$$

where the preconditioning matrix M^{-1} is chosen such that the spectrum of $M^{-1}S$ is more favorable than the spectrum of S leading to faster convergence, i.e., fewer CG-iterations, and such that the preconditioner step, i.e., $M\mathbf{z} = \mathbf{r}$, can be solved against low costs.

Algorithm 1 PCG algorithm (cf. [3]; Algorithm 9.1).

```
 $\mathbf{r} = \mathbf{b} - S\vec{\psi}$ , solve  $M\mathbf{z} = \mathbf{r}$  for  $\mathbf{z}$ ,  
 $\rho_1 = \langle \mathbf{r}, \mathbf{z} \rangle$ , set  $\mathbf{p} = \mathbf{z}$ .  
While (not converged)  
   $\rho_0 = \rho_1$   
   $\mathbf{q} = S\mathbf{p}$            Matrix-vector product  
   $\sigma = \langle \mathbf{p}, \mathbf{q} \rangle$        Inner product  
   $\alpha = \rho_0 / \sigma$   
   $\vec{\psi} = \vec{\psi} + \alpha \mathbf{p}$      Vector update  
   $\mathbf{r} = \mathbf{r} - \alpha \mathbf{q}$        Vector update  
  Solve  $M\mathbf{z} = \mathbf{r}$          Preconditioner step  
   $\rho_1 = \langle \mathbf{r}, \mathbf{z} \rangle$        Inner product  
   $\beta = \rho_1 / \rho_0$   
   $\mathbf{p} = \mathbf{z} + \beta \mathbf{p}$        Vector update  
End while
```

It was found that the RRB-method [4] is a very good preconditioner. RRB stands for "Repeated Red-Black" which refers to how nodes in a 2D grid are colored and numbered, see Figure 1. RRB- k means that the procedure is applied on k levels.

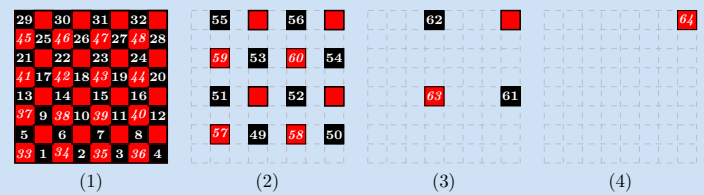


Figure 1: RRB-4 numbering for an 8×8 -grid. In each level first the black nodes are numbered, than half of the red nodes.



Basically, the RRB-method makes an incomplete factorization

$$S = LDL^T + R, \quad (5)$$

where L is a lower triangular matrix with unitary diagonal elements, D a block diagonal matrix, and R a matrix of adjustments resulting from so-called lumping procedures which eliminate fill-in in matrix L . As preconditioner for CG the matrix $M := LDL^T \approx S$ is taken. The resulting linear solver is called the RRB-solver.

In Figure 2 sparsity patterns of S and $L + D + L^T$ are shown. Note that within each red block the nodes do not depend on each other, which means that for each block computations can be performed in parallel.

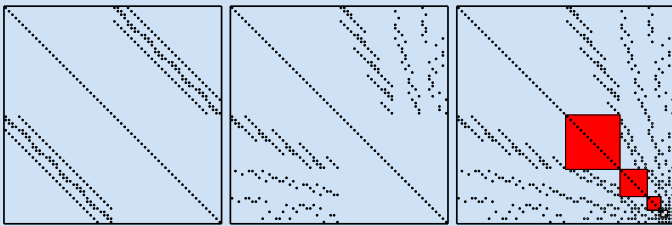


Figure 2: Sparsity patterns for $S \in \mathbb{R}^{64 \times 64}$. Left: matrix S with basic Red-Black numbering (RRB-1). Middle: matrix S with RRB-4 numbering. Right: matrix $L + D + L^T$. The red areas indicate where fill-in has been eliminated.

The RRB-solver is very efficient in itself which can be seen from the small number of CG-iterations needed for convergence. For all realistic test problems it was found that on average only 6 or 7 iterations are needed for convergence.

A very efficient implementation on the GPU

To get a really fast solver, the RRB-solver was implemented on the graphics processing unit (GPU) with CUDA. A GPU is suited for massively parallel computations as it contains thousands of processors following the single instruction multiple data (SIMD) architecture. A simple GPU offers already much more computational power than a high-end CPU. However, programming on the GPU comes with a rich set of rules. For example, it is very important that memory accesses are performed in a coalesced manner. Figure 3 shows how throughput deteriorates when this rule is not obeyed.

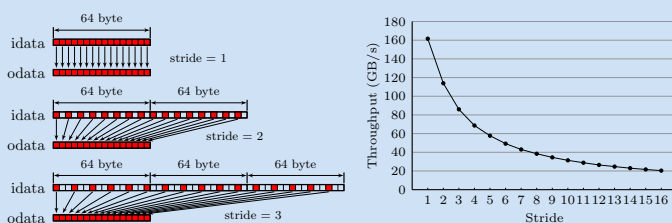


Figure 3: Accessing data with a stride is bad for throughput.

As the RRB-solver works with multiple levels, with a standard storage format throughput would be very low. For example, if on the first level only the red nodes were needed (stride = 2), then the throughput would already be reduced by 30%, and for coarser levels (stride = 4, 8, 16, ...) the throughput would drop dramatically, see Figure 3. Therefore, a different storage format was used, called the $r_1/r_2/b_1/b_2$ storage format [5], see Figure 4. The overhead of restoring data in the $r_1/r_2/b_1/b_2$ storage format is very little: only 2 vectors need to be restored at the beginning of CG. During the CG-iterations the storage format can be used for free, leading to maximal throughput on all levels.

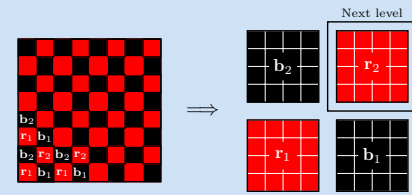


Figure 4: The $r_1/r_2/b_1/b_2$ storage format. The nodes are divided into four groups. The next coarser level is formed by the r_2 -nodes.

Experiments and results

The solver was tested with a collection of test problems, including Plymouth Sound (UK) and the IJssel (NL). In Figure 5 an example of the solution of Equation (1) is shown.

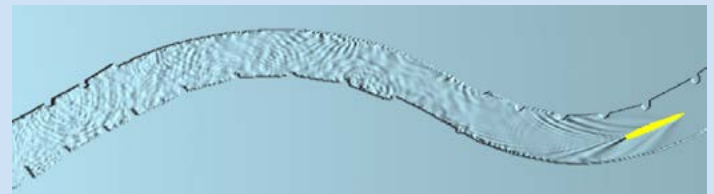


Figure 5: A ship sailing through the IJssel.

The experiments were performed in single-precision on a Dell T3500 workstation equipped with a fast Xeon processor and a GeForce GTX 580 graphics card. It was found that the largest test problems (1.5 million nodes) can be solved within 10 milliseconds, thus allowing real-time simulation. In Figure 6 speed up numbers are shown for the new CUDA RRB-solver versus a highly-optimized sequential C++ version which was used previously. On average the ship simulator is $25\times$ faster, so now $25\times$ larger domains can be simulated in real-time.

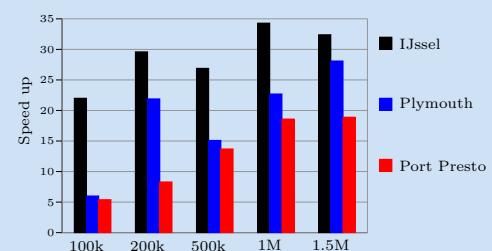


Figure 6: Speed up numbers for realistic test problems.

My special thanks go to professor Vuik and Auke Ditzel for their guidance and support during the project. For more information on this project you can always contact either of us.

References

- [1] A. Ditzel and N. Leith. Deep water anchor handling simulation. MARSIM, Singapore, 2012.
- [2] G. Klopman. Variational Boussinesq modelling of surface gravity waves over bathymetry. PhD thesis, University of Twente, 2010.
- [3] Y. Saad. Iterative methods for sparse linear systems. SIAM, 2nd edition, Philadelphia, 2003.
- [4] C. Brand. An incomplete-factorization preconditioning using repeated red-black ordering. Numerische Mathematik, 61, 433-454, 1992.
- [5] M. de Jong, A. v.d. Ploeg, A. Ditzel and C. Vuik. Real-time computation of interactive waves using the GPU. 15th NuTTS, Cortona, Italy, 2012



Enumeration and Design of Weighted Voting Games

Tomas Klos

Many decisions in groups of agents are made through ‘weighted voting’: each agent’s vote carries a *weight*, and if the sum of the weights of the agents who vote in favor of a proposal is at least equal to a certain *quota*, then the proposal is accepted. Sometimes, we want different agents to have different amounts of power in influencing the outcome of decision making. The following question then arises: How should we set the weights and the quota to achieve the desired distribution of power?

Decisions in the Council of Ministers of the European Union are made by *qualified majority voting*.¹ In the early years (1958–1973), there were six founding member states, that differed strongly in population size, so they received different ‘weights’ in the voting procedure: The votes of Italy, Germany, and France were each given a weight of 4; Belgium and The Netherlands 2; and Luxembourg 1. A proposed act was adopted if the votes of the countries in favor summed to at least the ‘quota’ of 12. A set of weights for the players plus a quota defines a *weighted voting game*.

We may ask: How does a country’s power in influencing the outcome of a decision depend on the weight of its vote? This depends on the definition of ‘power’ one adopts, of course. A commonly used *power index* was named after John Banzhaf [2]: Agent i ’s power is measured as the number of *losing* coalitions of *other* players that agent i can turn into *winning* coalitions by joining them. For example, there are four losing coalitions that The Netherlands can turn into winning coalitions by joining them, such as the losing coalition of Germany, France, and Belgium. See if you can determine Luxembourg’s Banzhaf power.

To formalize matters, if there are n agents, and we call the set of agents N , then player i ’s (normalized) Banzhaf index β_i is defined as follows.

$$\beta_i = \frac{\beta'_i}{\sum_{j=1}^n \beta'_j}, \quad \text{where} \quad \beta'_i = \sum_{C \in 2^{N-\{i\}}} (v(C \cup \{i\}) - v(C)).$$

The value-function $v : 2^N \rightarrow \{0, 1\}$ assigns a 0 to a losing coalition and a 1 to a winning coalition, where 2^N is the *power set* of N . The value β'_i is called player i ’s *raw Banzhaf index*. In the EU example, the raw Banzhaf index of the three weight-4 countries (FR, GE and IT) is 10, and of the two weight-2 countries (NL and BE) it’s 4 (and as you have just computed, for LU it’s 0!), so the normalized Banzhaf indices are 10/38, 4/38, and 0, respectively. (Note that, by definition, all agents’ normalized Banzhaf indices sum to 1.) We see that the *weight* of an agent’s vote and the agent’s (Banzhaf) *power* are not necessarily the same, and not even proportional.

The EU example raises the following question: If we want to have a certain desired or ‘target’ distribution of power over the agents (for example, the—perhaps intended—EU distribution of 4/17 for FR, GE and IT, 2/17 for BE and

NL and 1/17 for LU), how should we assign the weights (given a quota) to design a game that yields this distribution (or one close to it)? We have already seen that assigning weights in proportion to the desired power distribution may not work. Previous work [1, 3, 6] has focused on ‘local search’ algorithms that start with a quota and some distribution of weights and iteratively adapt these weights based on a comparison between the resulting game’s power distribution and the target distribution: The *weight* of a player whose *power* is too high is adjusted downward, and vice versa.

The main problem with these previous approaches is that they can’t guarantee that there does not exist a game with a power distribution *closer* to the target than the game they return. This is because these algorithms manipulate ‘weighted representations’ of WVGs. For example, the weighted representation of the EU-game is [12; 4, 4, 4, 2, 2, 1]. The problem is that there are *infinitely many weighted representations*, while there are only *finitely many games*. To see this, consider that every weighted voting game assigns a value of 0 or 1 to every coalition of agents. With n agents, this means there are at most 2^{2^n} games. It may thus be the case that in some iteration, such an iterative algorithm changes the players’ *weights*, while it doesn’t change the resulting *game*. Then in the next iteration, the power distribution also doesn’t change, and the algorithm stalls.

In our work, in contrast, we wanted to develop an *exact algorithm*, one that is *guaranteed* to return an optimal answer: a weighted voting game (in weighted representation) where the power distribution is *as close as it can be* to the target distribution. Note that the existence of a game with a power distribution *exactly equal* to the target distribution is not guaranteed. As we have seen, there are only finitely many games, and thus finitely many distinct power distributions, but there are infinitely many target distributions we can come up with.

Enumerating Weighted Voting Games

The approach we adopt, when given a target power distribution over n agents (a vector of n positive numbers summing to 1), is to **enumerate** all weighted voting games with n agents, **calculate** the power distribution for each enumerated game and its distance to the target (according to some distance measure), and **remember** the game with the power distribution closest to the target. While this seems easy enough, it’s not immediately obvious how we can enumerate all WVGs with n agents, although it is clear that we can’t enumerate games in their weighted representation, since there are infinitely many of those. We *can* enumerate all 2^{2^n} so-called *simple games*, but we will limit ourselves to *monotonic games*, which are games in which all supersets of winning coalitions are also winning coalitions: In a monotonic game, adding a player to a winning coalition can’t turn it into a losing coalition—a plausible restriction. In this context, it makes intuitive sense to talk about a game’s *minimal winning coalitions* (MWCs): An MWC is a winning coalition that becomes losing whenever some player is removed from it, and every superset of an MWC is also a winning coalition. Monotonic games can be represented as sets of minimal winning coalitions.

Every weighted voting game is a monotonic game, because weights are positive numbers: Adding a player’s weight to a winning coalition’s summed weight ($\geq$

¹See <http://wisv.ch/euvoting>.



q) can't decrease the coalition's weight below q . However, not all monotonic games are weighted: It's not always possible to construct a set of weights and a quota that yield precisely the value-function defining a monotonic game. To see this, consider the monotonic game G of four players with MWCs $\{1, 2\}$ and $\{3, 4\}$. Suppose, for the sake of contradiction, that there exist weights $w_1, \dots, w_4$ and a quota q that form a weighted representation of this game. Then $w_1 + w_2 \geq q$ and $w_3 + w_4 \geq q$, so $w_1 + w_2 + w_3 + w_4 \geq 2q$. But coalitions $\{1, 3\}$ and $\{2, 4\}$ are losing, since they are not (supersets of) MWCs, so $w_1 + w_3 < q$ and $w_2 + w_4 < q$, which means that $w_1 + w_2 + w_3 + w_4 < 2q$: A contradiction!

In our work, we use a further restriction, to the class of *canonical* weighted voting games. A canonical WVG (cWVG) is a WVG with a non-increasing order of weights in its weighted representation. For example, $[12; 4, 4, 4, 2, 2, 1]$ is a weighted representation of a cWVG. This is not really a restriction, because we can easily transform every WVG into a cWVG by just rearranging the players. Using the class of canonical WVGs *does* offer a significant benefit, that we can exploit in an enumeration algorithm: We have proven that there exists a particular *structure* in the class of cWVG's. Here we use the fact that every WVG, being a monotonic game, can be represented as a set of MWCs.

Theorem 1. Consider the (non-empty) set W of minimal winning coalitions of a canonical WVG. There exists a coalition $C \in W$ such that $W - \{C\}$ is the set of MWCs of another canonical WVG.

The proof can be found in [5] (an extended version of our conference paper [4]), a preprint that we recently submitted to a journal. Note that, although the theorem guarantees the existence of *at least* one such MWC in the set, there may be multiple such MWCs, so that each game may have multiple 'parents'. This is apparent in Figure 1, which shows all 27 canonical WVGs with four players. (We represent coalitions in *characteristic form*: a '1' indicates that the corresponding player is in the coalition, and a '0' that the player is not in the coalition: For example, the string '0011' represents the coalition containing players 3 and 4.) Each node in the graph represents a canonical WVG. The top node is the ('degenerate') game with *no* minimal winning coalitions, so all coalitions are losing. Looking at a node, a child represents a game whose set of MWCs contains the parent's MWCs, *plus* the coalition labeling the child: For each non-root node, this is the coalition C mentioned in the theorem. Thus, for example, the lowest coalition labeled '0110' in the second-to-left branch has as its set of MWCs $\{1100, 1010, 1001, 0110\}$. Both 1001 and 0110 can be removed from this set to yield sets of MWCs representing canonical WVGs.

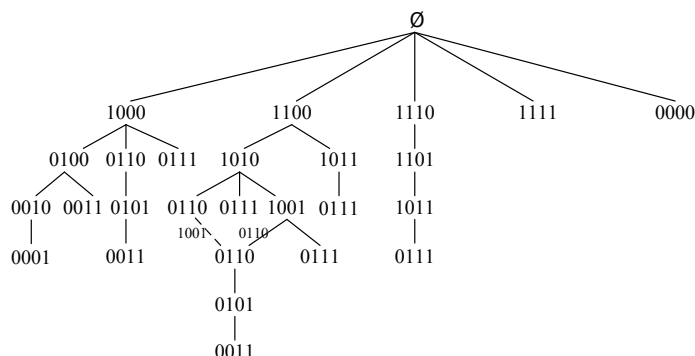


Figure 1: All canonical WVGs on four players.

Although this structure is nice, it is of little direct help in enumerating WVGs. It just says that for every canonical WVG with a non-empty set of MWCs, there exists some other canonical WVG—in other words, that in a graph like Figure 1, there is at least one path from every node to the root, basically. Fortunately,

we have also been able to 'turn Theorem 1 upside down,' as it were: We have a sure-fire way to find all children of every cWVG. Here, we use the notion of a *maximal losing coalition* MLC, which is defined analogously to a minimal winning coalition, and of the *i-th right truncation* function $\text{rtrunc}(C, i)$ of a coalition C , which returns a coalition with the i right-most players removed from C . For example, the 2nd right truncation of coalition 011101 is coalition 011000.

Theorem 2. If G and G' are n -player cWVGs such that G' is the child of G , and W (L) and W' (L') are G and G' 's sets of minimal winning (maximal losing) coalitions, respectively, then there exists a coalition $C \in L$ and an $i \in \mathbb{N}$ ($0 \leq i \leq n$), such that $W' = W \cup \{\text{rtrunc}(C, i)\}$.

(Again, the proof is in [5].) Now, at last, we have an exact algorithm for our problem. Given a target power distribution, we start with the canonical WVG with $\emptyset$ as its set of MWCs W . Then we check, for each coalition C in this game's set of MLCs (for the root-node this contains only the coalition with all n players) and every value of i between 0 and n , whether $W \cup \{\text{rtrunc}(C, i)\}$ is the set of MWCs of a canonical WVG. (This can be done efficiently using an algorithm by [7], which also gives us a weighted representation of this WVG.) For the top node, these games can be seen in the second row of Figure 1, for $i = 3, 2, 1, 0, 4$, respectively, from left to right. If this is true, we check whether we have already produced this game as the child of some other game, and if not, we have found a new canonical WVG. We compute the distance of its power index from the target, and if this is smaller than the smallest distance so far, we remember this game. We iterate from every new game we find. After enumerating all games, we output (a weighted representation of) the game that minimizes the distance to the target.

The work reported here was done by Bart de Keijzer in his MSc thesis from 2009. Bart was EWI's best MSc student in the year 2009-2010. He is now a PhD student at the Center for Mathematics and Computer Science (CWI) in Amsterdam.

References

- [1] Aziz, H., M. Paterson & D. Leech, "Efficient Algorithm for Designing Weighted Voting Games." *Proceedings IEEE Multitopic Conference*, 2007.
- [2] Banzhaf III, John F., "Weighted voting doesn't work: A mathematical analysis," *Rutgers Law Review* 19: 317-343, 1965.
- [3] Fatima, S., M. Wooldridge & N. Jennings, "An anytime approximation method for the inverse Shapley value problem." *Proceedings AAMAS*, 2008.
- [4] De Keijzer, B., T. Klos & Y. Zhang, "Enumeration and Exact Design of Weighted Voting Games." *Proceedings AAMAS*, 2010.
- [5] De Keijzer, B., T. Klos & Y. Zhang, "Solving Weighted Voting Game Design Problems Optimally: Representations, Synthesis, and Enumeration." *Arxiv preprint*, <http://arxiv.org/abs/1204.5213>, 2012.
- [6] Laruelle, A. & M. Widgrén, "Is the allocation of voting power among EU states fair?" *Public Choice*, 94: 317-339, 1998.
- [7] Peled, U. & B. Simeone, "Polynomial-time algorithms for regular set-covering and threshold synthesis." *Discrete Applied Mathematics*, 12: 57-69, 1985.



Photo by Eelco Visser

LabBack: extendible robust automatic assessment of student programs

Vlad Vergu

Coverage

Covering all areas from smartphones, portable music players, cooking robots to Internet core-infrastructure, certificate registrars and nuclear power plants, *software* continues to increase its infiltration in all aspects of our modern societies. Software is such a fundamental day-to-day presence that on the one hand the software engineering field appears largely unaffected by the current global recession. On the other hand software is so widespread that governments have finally become aware of the dangers, besides the opportunities, that it creates.

Dangers posed by software typically trace back to its poor quality, which in turn stems from the inadequate application of software standards and the intrinsic difficulty of building high quality software. At the very core of the problem lies a shortage of good software engineers in the marketplace, caused both by a general lack of engineers and an abundance of insufficiently educated ones.

However, engineers become good engineers through lots and lots of deliberate practice. Malcolm Gladwell claims that 10,000 hours of practice seems to be the magic number for producing brilliant experts, citing such gurus as Bill Gates of Microsoft, Bill Joy of Sun Microsystems, Steve Jobs of Apple, Eric Schmidt of Google and The Beatles. They all had two things in common: a favorable context for quickly becoming domain experts and the fact that they fought their way to privileged unlimited access to practice material.

There are three unfortunate discrepancies between Gladwell's experts and today's software engineers. Firstly, 10,000 hours is likely insufficient for today's engineers that do not benefit from the software greenfield of the 70s. Secondly, in today's dependence on software engineers, the industry cannot idly wait for the one in a million student who deliberately works his way to becoming a brilliant engineer. In other words: engineers are no longer pushed into the marketplace, but are pulled into it. Finally, today's students have less access to practice materials relative to the complexity of the software domain as compared to the 70s.

In fact students do not suffer from the nonexistence of practice material. Instead they suffer from the inability of their universities to provide this material and to train and evaluate their engineering skills. This is why the industry's promise of quicker and better training is perhaps true: they have a seemingly endless supply of practice material and teams to create continuous learning contexts.

The blocking factor in the universities' efforts to give plenty of practice materials is, surprisingly, time. Evaluating and commenting on students' solutions requires many man-hours. Significantly increasing the amount of practice material therefore quickly leads to implausible grading workloads. The obvious solution is to automatically evaluate (grade) students' solutions. Many automatic graders exist, and have since the '60s, but they all unfortunately exhibit at least one of the following properties:

- support only a single programming language
- could crash when grading certain programs
- could grade non-deterministically
- are susceptible to fraud
- do not scale up
- support only a single evaluation method

Since the coverage of requirements by existent solutions is incomplete an educator is forced to either develop yet another incomplete automatic grader or resort to manual grading. My Master of Science thesis presents 'LabBack', which aims to alleviate the above restrictions by providing a framework that ensures safe, deterministic and secure execution of students' programs, while being extendible with support for new programming languages and grading methods. Two revisions are presented, LabBack 1.0 and LabBack 2.0. The former addresses the safety, determinism and security requirements, the latter builds on that experience to ensure language and grading method extendibility.



LabBack 1.0

LabBack 1.0's design and implementation form an experiment aimed at demonstrating the feasibility of automatically evaluating runtime correctness of students' programs in a secure, robust and timely manner. The approach is to run all student programs within a single instance of the Java Virtual Machine (JVM). Thus each program as a separate thread within a daemon instance of LabBack. As new programs arrive from LabBack's client - in this case WebLab, developed by Dr. Eelco Visser - the processing of these programs is scheduled on the available resources and subsequently performed. Scheduling is of course required to prevent overloading the host machine when more programs arrive than the number of available CPUs.

To prevent resource hogging by unresponsive programs LabBack continuously monitors guest programs and automatically terminates their respective threads after a certain timeout. The forcible termination of hung programs is achieved by calling the deprecated 'stop' method on the hosting thread. This method is deprecated in the Java Runtime because as a thread dies it relinquishes all locks it holds on objects, thus exposing objects in possibly invalid states to other threads.

This deprecation highlights in fact two serious issues. Firstly, since a guest program is a single thread, for a thread to hold locks on other objects than its own it would imply either its accessing of LabBack's control threads or of other guest programs' threads. Both cases must of course be forbidden in order to prevent fraud attempts. Secondly, a Java thread always has access to certain objects that it has not loaded itself that, namely instances of the Class class. Java's class loading mechanism is thoroughly described in the Java Virtual Machine Specification, but in a nutshell classes are identified by their qualified names, loaded lazily and subsequently retrieved from cache. The fact that a class is only loaded once leads to the possibility that one of two guest programs that both declare a class identified by the same qualified name will use the other program's class instead of its own. To alleviate these problems LabBack loads each guest program within its own insulated class loader. This class loader ensures that classes pertaining to the Java Runtime are loaded by its parent class loader (to prevent fraudulent overriding of built-in classes) and that students' classes are cached locally. The latter property allows the guest classes to be unloaded together with the student program thus preventing memory leaks in the permanent generation space of the JVM. LabBack uses the Java Security Manager to prevent guest programs from reflectively obtaining references to objects whose classes have not been loaded by the guest program. These measures effectively guarantee that guest programs' threads can never hold locks on objects owned by other threads and thus ensure security and determinism of execution and allow threads to forcibly stopped without any risks to the rest of the JVM.

LabBack 1.0 was evaluated during the Concepts of Programming (TI1220) course of 2011-2012 where it was used as the automatic grading system for solutions submitted through WebLab. By the end of the course results from more than 49,000 automatic evaluations of programs written in Scala, C and Javascript were used in grade calculations. LabBack 1.0 is estimated to have executed and evaluated more than 2,000,000 programs.

LabBack 2.0

Building on the empirically proven model of LabBack 1.0, LabBack 2.0 aims to be extendible with new programming language and grading criteria support. The resulting application is a hybrid between a traditional middleware web application and a plugin framework. LabBack 2.0 combines the prototype's secure runtime model with dynamic composition of program execution pipelines to form an extendible platform for hosting secure evaluation of stu-

dents' programs. This platform is extendible with new language-specific and experiment-specific program evaluation techniques by means of cooperating plugins. Its design allows it to scale across multi-instance deployments and employs load distribution techniques that balance execution time with cost.

LabBack 2.0 is a web application that runs on Apache Tomcat. Language-specific and experiment-specific functionality is provided by LabBack plugins which are dynamically loaded and their executing or evaluation functionalities are composed based on each guest program's requirements. This allows LabBack to support evaluation in various programming languages and techniques within single instances. An approach similar to per-program insulated class loaders is applied for plugins to allow plugins to be unloaded when they are no longer needed, thus reducing the nominal memory footprint of the application. The plugin extension mechanism allows educators to provide course-specific support without having to worry either about security or about scalability.

Since a single hosting machine may have insufficient resources to support very popular courses, LabBack 2.0 needs scale past single machines as needed. Since most current Virtual Private Server (VPS) hosts bill on a VPS uptime basis keeping many idle VPS instances running is cost inefficient. LabBack thus needs to trim the number of server instances based on the experienced load. Traditional load balancing mechanisms are targeted at equal balancing of loads across multiple servers and if they are capable of packing loads in clusters the loads are estimated by the rate of incoming TCP/UDP requests, which is at best a poor indicator. LabBack deployments are forest-shaped and load distribution instead uses a subtree load heuristic based on job queue length and age called 'tree badness' (described in the thesis). The tree badness is calculated at each processing node and is a factor of the tree badnesses of every descendant node and of the capacity, queue length and queue age of the calculating node. The tree badness measure is propagated towards ancestor LabBack nodes by piggybacking it on messages containing results of program evaluation. Since the heuristic calculated at a single node takes into account measurements for all the descendants of that node it gives a numeric estimate of the load for that entire subtree; estimate which is affected by the subtree height and capacity. Any LabBack instance can choose whether to evaluate a program locally or on one of its descendant nodes. By tuning of the tree badness function and by configuring a threshold value, system engineers can balance between depth-first or breadth-first subtree activation.

Conclusion

LabBack 2.0's extension model was evaluated by reimplementation of the Scala, Javascript and C grading modules from LabBack 1.0 as plugins. The security and robustness characteristics were evaluated by re-running all of the 49,000 evaluations from the TI1220 course and comparing with the initial results. This latter evaluation confirmed that LabBack 2.0 preserves the initial functionality but highlighted that the unit testing approach used for grading of C programs was sometimes non-deterministic, biased in the students' advantage, thus highlighting the need for careful rethinking of how C programs should be graded.

The scalability of LabBack 2.0 deployments, and therefore the suitability of the tree badness heuristic, was evaluated by simulation in a variety of network topologies. Results have shown that tree badness is a good metric for load packing in a forest, that it allows the number of alive nodes to convergence quickly but that good threshold values depend on deployment depths. This latter finding is explained by the multiplicative shape of the tree badness calculation and leads to the recommendation that LabBack deployments have mostly balanced subtree heights. 



Inside the INSYGHT Lab

Cees-Willem Hofstede, Christiaan Kehl, Thoman Kroes, Ricardo Lopes, Noeska Smit, Tim Tutenel

On the 12th of November last year, the INSYGHTLab was officially opened. This lab provides a common space for computer science researchers and students to perform experiments and do projects in several different, but closely related fields. In this article we take a closer look at some of the projects that are worked on in this lab involving the Computer Graphics and Visualization group.

On the 2nd floor of our faculty, behind the mysterious doors with the INSYGHT-Lab logo, researchers and students alike work on their projects. You may have already seen a part of the lab in a recent Tweakers videoreport [1]. This article takes a closer look at the projects currently taking place in the INSYGHTLab involving the Computer Graphics and Visualization (CGV) group.

The INSYGHTLab provides a space for experimental setups of various research fields within the department of Intelligent Systems. The sections Computer Graphics and Visualisation, Interactive Intelligence, Multimedia Signal Processing and Pattern Recognition and Bioinformatics all share the INSYGHTLab to collaborate on different projects. Both researchers and students have the space and equipment they need set up there to work on their projects.

In this article we will describe the following projects: Exposure Render: photorealistic rendering of medical volumes, 3Di: visualization of flooding simulation, Bicycle Up: a serious bike-riding game and DOKJay: a serious Microsoft Surface Table game.

Exposure Render

In the field of medical visualization one of our current projects is Exposure Render. Based on image data from CT-scanners and volume rendering techniques, the open source Exposure Render application allows users to render realistic-looking images interactively. A CT image volume consists of tissue density values and by assigning these values to a colour and opacity a 3D image can be constructed. Using GPU programming techniques, physically-based light transport calculations are applied to these 3D volumes resulting in realistic-looking images. To speed up the process and keep interaction with the volumes real-time, Monte Carlo ray tracing (MCRT) is applied. This technique renders a scene by randomly tracing samples of possible light paths. By repeating this often for every pixel and averaging the results the correct solution will appear.

The INSYGHTLab has a powerwall (a large high-resolution display wall) available where we can give live demos of the software to visitors. If you have a nice NVIDIA GPU in your computer at home, Exposure Render is freely available to play around with [2].

3Di

In our 3Di project [4], we connect research on rendering of massive point clouds with the visualization of large-scale, realistic flooding simulations.



Figure 1: Example of Exposure Render rendering of a CT-scan of the head.

3Di is a system that is designed to enable interactive discussions of flood scenarios between water boards, decision makers, hydrologists and further water experts. The open AHN-2 (Actueel Hoogtebestand Nederland) dataset is our foundation for the visualization [5]. This is a full-scale aerial laser scan of the whole Netherlands in a resolution of 30 points/m². Due to the spatial extent and resolution of the dataset, we search for new Level-of-Detail rendering algorithms to interactively display these Terabytes of data.

This geographic dataset can successively be used in Visual Analytics to study a diversity of climate change phenomena. Our project focuses on the study of flood events by combining our AHN-2 rendering with the visualization of Sobek simulation- and Subgrid-method output. First results show that this kind of visualization helps in understanding the behaviour of historical floods (like the North Sea Flood of 1953) as well as in discussing new flood protection mechanisms and evacuation policies. To support this analytical process, particularly in urban environments, 3Di is able to render the scene in 3D stereo mode. For this purpose, a workbench with a 3D projection surface of 1.80m*1.10m is used in the INSYGHTlab (figure 2). More information on the projects can be found at our project page [3].

Bicycle Up

At the INSYGHTlab we are also focused on serious games, i.e. games with purposes other than fun. Our Bicycle Up project aims at teaching children and teenagers with Down Syndrome to ride a bicycle, in a safe environment. In the Netherlands, children with Down Syndrome are, like everyone else, highly motivated to ride bicycles. Among other advantages, this can allow them to be more independent. However, these children have more difficulties than the rest of us. The sensory overload that comes with riding a real bicycle leads to stress, fear, lack of practice and to the recurrent need to re-learn it.



Figure 2: 3Di: Working the workbench (picture by Annelies te Selle)



Figure 3: DOKJay game running on the Microsoft Surface Table

We are combining a physical setup with a computer game which helps these children to experience how to ride a bicycle. Using game design techniques, we are developing an immersive game which not only makes this learning process fun and empowering, but decreases the associated stress and fear.

This project is a joint effort with the Bicycle Dynamics group (3mE faculty). A mechanical static bicycle controller is being developed for the game, which allows players to steer and pedal like they would do in real life, but controlling the in-game virtual bicycle. This controller implements a mathematical model, which captures the mechanics and dynamics of balancing a bicycle.

By using this setup, we have the control to tweak the model parameters. Among other things, we can make it easier or more difficult (i.e. closer to reality) to balance a virtual bicycle. In a game, this technique has the power of dynamically changing how stable a virtual bicycle is, thus how easy it is to balance it. With such a mechanism we can adapt the game to gradually and better teach the player, in the safest and most personal way possible. Bicycle Up has already caught the attention of various parties, including an international commercial partner.

DOKJay

Another project carried out in the lab is DOKJay. This serious game commissioned by the Delft public library DOK originated from the master course 'Building Serious Games', lectured by Rafael Bidarra. The people of DOK wanted

a novel way to attract youngsters to the library. They know music is a great means to reach their target group and have the right facilities. A Microsoft Surface Table offering a multi touch enabled environment was considered perfectly suitable to form the core of the game.

After some brainstorming and planning, seven students from EEMCS and TPM faculties started their journey to bring this game to life. Although the table is not all that easy to set up, after some tweaks and new hard disks, a completely revised setup was realized. Thanks to these modifications, the game could run on the older surface tables as well as on the newer models recently introduced.

With four musicians, the team was able to come up with an attractive game that invites to listen to music, to interactively explore musical structures and is foremost a lot of fun (figure 3). It has two modes, a puzzle mode and a free mode. In the puzzle mode the player can select a musical style. Then a song in that style is cut up into fragments, shown in random order. The player has to reorder the song parts by listening carefully to each piece. The current prototype version contains the classical piece 'Canto Ostinato', as well as the ever so popular 'Gangnam Style'. A rock classic should not be missing, so a song by 'Queens Of The Stone Age' is available in level tree.

Whenever a puzzle is solved, samples of musical instruments are unlocked, which the player can use in the free mode to create his or her own music composition. So by solving more puzzles, you get more freedom to unleash your musical potential in free mode. When a song is completed you can share it over social media by scanning a code with your phone.

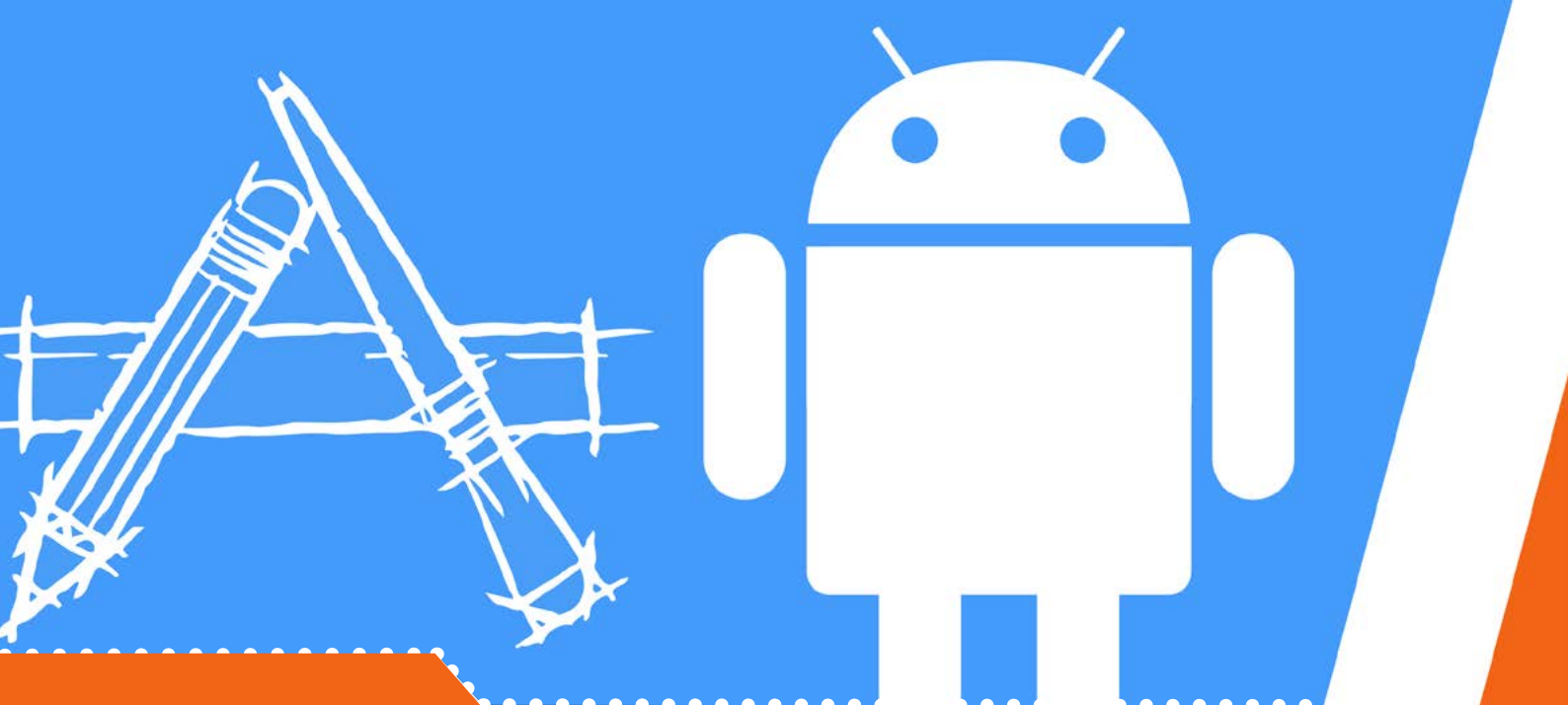
The DOKJay project successfully shows how game technology, media and online features can be combined in a socially relevant context.

More information

More information about the lab and its interesting projects can be found at the INSYGHTlab website [6]. If you'd like to work on an interesting project for your bachelor or master thesis with the CGV group, please contact Rafael Bidarra or Elmar Eisemann. 

References

- [1] <http://tweakers.net/video/6877/insyghtlab-visuele-proeftuin-van-tu-delft.html>
- [2] <http://code.google.com/p/exposure-render>
- [3] <http://graphics.tudelft.nl/Projects/FloodVis>
- [4] <http://www.3di.nu>
- [5] <http://www.ahn.nl>
- [6] <http://insyghtlab.tudelft.nl>



Mobile Apps: de ene app is de andere niet

DSW

Mobile development heeft in de afgelopen vijf jaar een enorme vlucht genomen sinds het uitkomen van de iPhone in 2007. Het maken van apps voor deze platformen is alleen niet eenvoudig door de grote verschillen tussen alle operating systems. Een iOS-app draait niet op Android, maar er zijn technieken om hieraan te ontkomen.

DSW Zorgverzekeraar vindt innovatie belangrijk, zowel qua ICT als op andere vlakken. We zijn bezig met het opzetten van een mobile app-team, omdat we veel zien in mobile als kanaal naar onder andere onze verzekerden. Dit kanaal zal ook steeds belangrijker worden! Daarom willen wij een mobile app-team oprichten, met de vrijheid van een kleine startup. In dit artikel geven we je graag onze visie op mobile app development, maar als jij ook een mening hierover hebt: we hebben nog stoelen vrij!

Architectuur-dierentuin

Als we kijken naar verschillende soorten apps, komen native apps als eerste naar boven, geschreven in de eigen taal van het operating system. Dit zijn de vaak bekende talen: Java voor Android, van Google, en Objective-C voor iOS, van Apple. Deze talen geven je de meeste controle over je omgeving en maken natuurlijk altijd optimaal gebruik van alle nieuwe APIs van je operating system, maar werken slechts op een platform. Dit betekent alle code twee keer uitschrijven, waarbij sommige bedrijven zelfs aparte teams voor verschillende platformen hebben. Herbruik van kennis is dus soms heel klein, wat eigenlijk zonde is.

De tweede mogelijkheid is het uitwerken van een web app, een website geoptimaliseerd voor mobiel gebruik. Het voordeel hierbij is dat hij (redelijk) zou moeten werken op meerdere platformen. Je kunt ontwikkelen in HTML, CSS en Javascript, talen waarmee meer ontwikkelaars bekend zijn. Nadelen zijn hier helaas ook: deze web app werkt meestal alleen online. Met HTML5 local storage

valt het een en ander op te lossen, maar dan nog heb je geen toegang tot eigen features van de telefoon, zoals de camera of GPS. Hoewel een web app dus prima kan werken, is het ook niet perfect voor alle situaties.

Een derde mogelijkheid sinds de start in 2009 is Phonegap. Dit platform biedt je de mogelijkheid je app te schrijven in HTML, CSS en Javascript, die dan in een fullscreen browser (de UIWebView) als app worden gedraaid. Speciale Javascript APIs bieden toegang tot de camera, GPS en dergelijke. Wil je meer, dan kun je native plugins aan je app toevoegen.

Een bekend probleem van Phonegap is de performance. Apps reageren vaak niet snel genoeg, zeker op oude iPhones of budget-Androids. De performance van HTML-apps in het algemeen is een van de redenen dat Facebook ook over is gegaan van een HTML app op een native app. De keuze van eventuele Javascript frameworks maakt ook uit voor de performance, sommige frameworks wachten na een touch event 300 milliseconden om te zien of het geen dubbel-tap wordt. Een vertraging van 300 ms is genoeg om de meeste gebruikers af te doen haken. Ook heeft iOS sinds kort de Javascript acceleratie, genaamd Nitro, in de UIWebView uitgezet. In Safari staat hij nog wel aan, de reden van het uitzetten is tot nu toe niet bekend. Dit heeft natuurlijk ook zijn effect op de performance van Phonegap!

De laatste categorie oplossingen om een app te schrijven is helaas ook niet zaligmakend. Dit zijn de 'Write-Once, Run-Anywhere' (WORA) oplossingen. Grofweg zijn er twee soorten oplossingen in deze categorie. Optie een is je code schrijven in een gezamenlijke taal, zoals HTML, CSS en Javascript, die daarna naar de native taal wordt gecompileerd. De tweede soort stuurt je non-native code met een interpreter naar de telefoon. Voorbeelden van deze laatste zijn Titanium Appcelerator of Xamarin. Phonegap van hierboven is dus eigenlijk ook zo'n oplossing. Dit betekent helaas wel dat je app groter wordt, omdat de interpreter meestal moet worden meegestuurd. Een ander nadeel van deze aanpak is dat je continu achter de feiten aanloopt als er een nieuwe API uitkomt voor de telefoon: nieuwe APIs moeten eerst voor jouw platform oplossing geïmplementeerd worden, door de ontwikkelaars van jouw oplossing, voor je ze zelf kunt gebruiken. Ook ben je gebonden aan die (vaak commerciële) toeleverancier.

HTML



Strategie?

Hoe maak je nou een mobile app die meerdere platformen kan raken? Bekend software engineer Martin Fowler onderscheidt twee soorten apps: producten en channels. Een product is iets dat klanten van zichzelf al willen, zoals veel spelletjes. Een channel app is een nieuwe manier van communiceren met je bestaande klanten, bijvoorbeeld de Facebook-app.

Als je een product maakt, wil je dat je app meteen alle mogelijke features aanbiedt. Als het niet in je app zit, kun je het namelijk niet aanbieden aan je klant. Tenslotte, User Experience is van levensbelang bij een mobile app: Een trage app krijgt slechte reviews en wordt niet meer gedownload. Dat moet vanaf het begin al goed zijn!

Een channel app kan gedeeltelijke functionaliteit aanbieden in het begin, omdat er ook andere kanalen zijn waardoor je met je klanten communiceert. Je wilt alleen zoveel mogelijk verschillende klanten met dit nieuwe kanaal bedienen. Zorg dus dat je snel alle platformen bedient, en bouw gaandeweg de functionaliteit uit. Mobile is voor DSW een nieuw channel, waar we veel kansen in zien en dat in de toekomst nog veel belangrijker wordt. We moeten dus meteen vanaf het begin alle platformen raken en gaandeweg functionaliteit toevoegen.

Best of both worlds?

We moeten dus als DSW tegelijkertijd voor meerdere platformen ontwikkelen, maar geen van de eerdere oplossingen is perfect. Kunnen we de voordelen van native niet combineren met de voordelen van HTML apps? Volledige controle over je app en razendsnelle performance, maar wel herbruik van code? Het is LinkedIn wel gelukt: de LinkedIn-app is voor naar schatting 95 procent geschreven in HTML, slechts een klein laagje van zo'n vijf procent is native. Deze vijf procent onderhouden is veel minder werk dan de hele app native maken, maar je hebt wel de mogelijkheid eigen features van je platform te gebruiken en bent niet afhankelijk van een toeleverancier. Dit betekent dat de look en feel van de app niet native is. Je ziet dat het geen volledig iOS of Android app is, maar dat hoeft geen probleem te zijn zolang het maar herkenbaar is als de DSW-app. Al met al combineert dit dus de voordelen van native en cross-platform!

Daarom hebben we voor deze oplossing gekozen. Ons mobile app-team is in oprichting en zoekt nog mensen, creatieve en ambitieuze universitaire beta-starters. Kennis van iOS, Android of HTML / CSS / Javascript zijn een pre, maar zeker niet vereist: we leiden je op tot volwaardig mobile app developer. Binnenkort gaan we van start! Kom jij ons team versterken?

DSW
zorgverzekeraar
goed voor je

Advertorial

Java Puzzlers

De Puzzler

Solution

Last MaCHazine you had the puzzle 'Class Warfare'. Now we present the solution of this puzzle.

A quick look suggests that the program should print physics chemistry biology; after all, Java loads classes at run time, so it always has access to the latest version of a class. A deeper analysis suggests otherwise. **References to constant fields are resolved at compile time to the constant values the denote** [JLS 13.1]. Such fields are technically, if oxymoronically, known as *constant variables*. A constant variable is defined as a variable of primitive type or type String that is final and initialized with a compile-time constant expression [JLS 4.12.4]. With the benefit of this knowledge, it would be reasonable to think that the client program compiles the initial values of Words.FIRST, Words.SECOND, and Words.THIRD into its class file and prints the null set, regardless whether the class Words has been modified.

Reasonable, perhaps, but not correct. If you ran the program, you found that it prints the chemistry set. This seems truly bizarre. Why would it do a thing like that? The answer is to be found in the precise definition of the term *compile-time constant expression* [JLS 15.28]. The definition is too long to reproduce here, but the key to understanding the behavior of the program is that **null is not a compile-time constant expression**.

Because constant fields are compiled into clients, **API designers should think long and hard before exporting a constant field**. If a field represents a true constant there is no harm in making it a constant field. If, however, you want clients to adapt to changes in the field, make sure that it isn't a constant. There is an easy way to do this: If you initialize a field, even a final field, with an expression that isn't constant, the field isn't constant. You can turn a constant expression into a nonconstant by passing it to a method that simply returns its input parameter.

If we modify the class Words to use such a method, PrintWords will print physics chemistry biology after Words is again modified and recompiled:

```
public class Words {
    private Words() { }; // Uninstantiable
    public static final String FIRST = ident("the");
    public static final String SECOND = ident(null);
    public static final String THIRD = ident("set");

    private static String ident(String s) {
        return s;
    }
}
```

Despite their name, enum constants, introduced in release 5.0, are *not* constant variables. You can add enum constants to an enum type, reorder them, and even remove unused enum constants without the need to recompile clients.

Puzzle

Each MaCHazine there will be a puzzle with a script of a program, here is the new one. It is up to you to find out what the program does. You may use Java, but it's more fun just to use your mind.

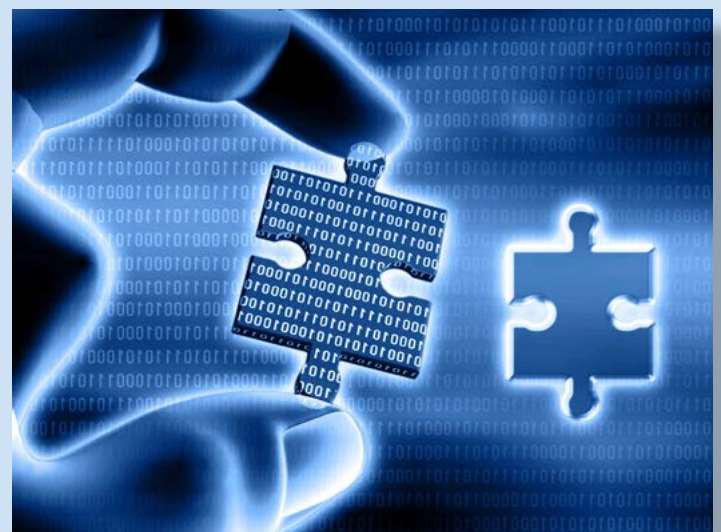
This program consists entirely of synchronized static methods. What does it print? Is it guaranteed to print the same thing every time you run it?

```
public class PingPong {
    public static synchronized void main(String[] a) {
        Thread t = new Thread() {
            public void run() { pong(); }
        };
        t.run();
        System.out.print("Ping");
    }

    static synchronized void pong() {
        System.out.print("Pong");
    }
}
```

References

- [1] Java puzzlers, traps Pitfalls and corner cases, Joshua Bloch & Neal Gafter





Wiskundepuzzel

Mark Veraar

Oplossing

Er waren vier inzendingen op de puzzel 61. De inzenders waren Rein Leentfaar (buiten mededinging), Romke Rozendaal, Jolien de Haas en M. Looijen. Zij gaven allen het goede antwoord. Elk van de inzenders verdient 5 punten voor de ladder.

We kunnen er veel woorden aan vuil maken, maar de oplossing spreekt voor zich.

$\times 10$		$+17$	$+10$		
5	2	4	6	3	1
$: 3$	$+9$				$+10$
1	5	3	2	6	4
		$\times 10$	$+5$		
3	4	5	1	2	6
$+10$					$+10$
4	6	2	3	1	5
$: 3$	-2	$\times 5$		$\times 20$	
6	3	1	5	4	2
		$+10$			
2	1	6	4	5	3

Wiskundeladder na puzzel LX

- | | | |
|----|----------------------|---|
| 1. | Jeroen Wille | 9 |
| 2. | Marieke van der Tuin | 8 |
| | Erik Ammerlaan | 8 |
| | Robbie Elbertse | 8 |
| 5. | Micke Vrolijk | 6 |
| | Leon Planken | 6 |
| | Jonatan Bijl | 6 |
| 8. | Jarno Hartog | 5 |
| | Jeff Smits | 5 |
| | Michiel de Reus | 5 |
| | Wim van Geloven | 5 |
| | M. Looijen | 5 |
| | Romke Rozendaal | 5 |
| | Jolien de Haas | 5 |

Puzzel

Beschouw het volgende gedachte-experiment. Ik verzin 100 verschillende gehele getallen en schrijf elk van deze getallen op en doe deze in verschillende enveloppen. De getallen kunnen erg groot zijn, maar hierover heb je verder geen informatie. De enveloppen schud ik goed door elkaar en leg ze op een stapel.

Nu mag jij het volgende programma uitvoeren:

1. Je opent de volgende envelop van de stapel. Het getal dat je ziet onthoudt je goed.
2. (Je kiest het getal van de laatst geopende envelop en stopt) óf (je gaat terug naar stap 1).

Indien je net zo lang doorgaat totdat de enveloppen op zijn, dan kies je het getal uit de laatste envelop.

De opdracht is om een strategie te vinden zó dat je met grote kans het grootste getal kiest. Let op: je mag alleen het getal van de envelop kiezen die je als laatste hebt geopend. Een uiterste strategie zou zijn: je kiest altijd de eerste envelop. In dat geval heb je natuurlijk maar 1% kans dat je het grootste getal hebt. Een ander uiterste zou zijn om altijd de laatste envelop te kiezen. Ook in deze situatie heb je ook maar een kans van 1% dat je het grootste getal hebt.

De precieze opdracht is als volgt:

- (a) Geef een strategie waarbij je met meer dan 35% kans het grootste getal vindt.
- (b) Wat kun je zeggen over de situatie waarbij we hetzelfde doen met 500 gehele getallen?
- (c) En wat gebeurt er indien we n gehele getallen nemen met n groot?



Het Keuzeaxioma

Aad Vijn

Inleiding

Sinds de opkomst van de moderne verzamelingentheorie, bedacht door Georg Cantor aan het begin van de twintigste eeuw, zijn wiskundigen op zoek gegaan naar een stevig fundament waarin de wiskunde op te bouwen is (later is gebleken dat dit te optimistisch is: we kunnen altijd maar een deel van de wiskunde opbouwen aan de hand van een axiomatisch systeem). Verschillende axiomatische systemen werden hiervoor geconstrueerd, waaronder het axiomatisch systeem van Zermelo en Fraenkel dat terug te vinden is in het college “Wiskunde Structuren”. We noteren dit systeem met **ZF**.

Een toevoeging aan dit systeem is het zogenaamde Keuzeaxioma. De rest van dit artikel zal hierover gaan. We zullen de volgende vragen kort de revue laten passeren: Wat is het Keuzeaxioma?, Waarvoor hebben we het nodig? en ten slotte behandelen we een maximaalprincipe dat equivalent is aan het Keuzeaxioma en een belangrijk resultaat. Uitgebreidere resultaten zijn terug te vinden in mijn scriptie “Het Keuzeaxioma: een studie naar de geschiedenis van het Keuzeaxioma, de gevolgen, equivalente beweringen en de onbewijsbaarheid” (2012).

Wat is het Keuzeaxioma

Ernst Zermelo was de eerste die het Keuzeaxioma formuleerde. Hij gebruikte zijn formulering in [3] om te bewijzen dat iedere verzameling te welordenen is. De formulering was als volgt. Als M een verzameling is, dan:

Jeder Teilmenge M' denke man sich ein beliebiges Element m' zugeordnet, das in M' selbst vorkommt und das “ausgezeichnete” Element von M' genannt werde möge. [E. Zermelo, 1904]

Een aantal jaren later werd het Keuzeaxioma geformuleerd in termen van *keuzefuncties*. We definiëren nu wat een keuzefunctie is.

Definitie 1 (Keuzefunctie). *Gegeven een familie niet-lege verzamelingen $\mathcal{F}$. Een keuzefunctie op $\mathcal{F}$ is een functie $f : \mathcal{F} \rightarrow \bigcup \mathcal{F}$ zó dat voor iedere $F \in \mathcal{F}$ geldt dat $f[F] \in F$.*

In woorden doet deze f precies wat we willen: uit iedere verzameling F kiest de functie een element uit de verzameling F . Nu is het Keuzeaxioma als volgt de formuleren:

Keuzeaxioma1. *Voor iedere familie $\mathcal{F}$ van niet-lege verzamelingen bestaat er een keuzefunctie.*

Een vaak gebruikte equivalente definitie van het Keuzeaxioma is in termen van het Cartesisch product.

Keuzeaxioma2. *Voor iedere familie van niet-lege verzamelingen $(F_i)_{i \in \mathbb{I}}$ geldt dat $\prod_{i \in \mathbb{I}} F_i = \{f : \mathbb{I} \rightarrow \bigcup_{i \in \mathbb{I}} F_i \mid \forall i \in \mathbb{I} (f(i) \in F_i)\}$ niet leeg is.*

Waarvoor hebben we het Keuzeaxiom nodig?

Waar hebben we het Keuzeaxioma eigenlijk voor nodig? Dit is een lastige vraag om te beantwoorden zonder dat men de precieze rol van het Keuzeaxioma in de wiskunde begrijpt. Termen als *consistentie* en *onafhankelijkheid* spelen een belangrijke rol uit de modeltheorie, maar naïef gezegd maakt het Keuzeaxioma het mogelijk om te kunnen rekenen in het oneindige.

Tijdens de opkomst van het Keuzeaxioma begonnen wiskundigen theorieën die door de afgelopen eeuwen heen zijn opgesteld opnieuw te analyseren om te zien hoe men te pas en te onpas gebruikmaakte van het Keuzeaxioma. Waar men het Keuzeaxioma vond werd vervolgens (vaak) een nieuw bewijs gegeven waarin het Keuzeaxioma niet werd gebruikt. Dit “muggenziften” is te danken aan Dhr. Luitzen Brouwer, die het *constructivisme* opstelde.

We illustreren het nut van het Keuzeaxioma aan de hand van de Neststelling van Cantor waar blijkt dat Keuzeaxioma nodig is in het bewijs (deze redenatie is eigenlijk een beetje krom: er is nog geen bewijs gegeven waaruit blijkt dat het Keuzeaxioma niet nodig is; zodoende nemen we aan dat het zonder het Keuzeaxioma niet kan). Allereerst formuleren we de stelling.

Stelling 1 (Neststelling van Cantor). *Zij (X, d) een volledige metrische ruimte. Laat $(F_n)_{n=0}^\infty$ een rij niet-lege gesloten deelverzamelingen zijn zodanig dat $F_0 \supset F_1 \supset F_2 \supset \dots$ en*

$$\lim_{n \rightarrow \infty} \text{diam}(F_n) = 0.$$

Dan geldt dat er precies één punt x_0 is, zó dat

$$\bigcap_{n=0}^\infty F_n = \{x_0\}.$$

Het bewijs van deze stelling maakt direct gebruik van het Keuzeaxioma. Voor alle $n \in \mathbb{N}$ kiezen we simultaan uit F_n een element dat we x_n noemen (zo verkrijgen we een Cauchy-rij die, door de volledigheid van de metrische ruimte, convergeert naar x_0). Zo’n greep uit een aftelbaar oneindige verzameling van niet-lege verzamelingen vereist het Keuzeaxioma.

Merk op dat de “Reële versie” van de neststelling van Cantor *niet* het Keuzeaxioma vereist. Deze versie zegt dat, voor een dalende rij gesloten intervallen

$$[a_1, b_1] \supset [a_2, b_2] \supset [a_3, b_3] \supset \dots$$

van $\mathbb{R}$ met $\lim_{i \rightarrow \infty} \text{diam}[a_i, b_i] = 0$, geldt dat er precies één punt x_0 is in de doorsnede $\bigcap_{i \in \mathbb{N}} [a_i, b_i] = \{x_0\}$. We kunnen dan uit ieder interval bijvoorbeeld het midden $c_i = \frac{a_i + b_i}{2}$ nemen. Via deze weg maken we expliciet een Cauchy-rij die het werkt doet; het Keuzeaxioma is dan dus niet nodig.

Equivalente beweringen

Het is misschien duidelijk dat met de definitie van het Keuzeaxioma niet echt te werken valt. Daarom zoeken we naar equivalente stellingen waarmee wel te werken is. Het *Lemma van Zorn* is zo’n equivalente stelling aan het Keuzeaxioma. We definiëren het Keuzeaxioma en laten een toepassing zien. Het bewijs dat het Lemma van Zorn daadwerkelijk equivalent is aan het Keuzeaxioma staat in mijn scriptie.



Het Lemma van Zorn

Zij P een verzameling. We noemen een relatie $\leq$ en *partiële ordening* op P als deze voldoet aan de volgende eigenschappen:

- (P1) Voor alle $p \in P$ geldt $p \leq p$;
- (P2) Voor alle $p, q \in P$ geldt: als $p \leq q$ en $q \leq p$ dan $p = q$;
- (P3) Voor alle $p, q, r \in P$ geldt: als $p \leq q$ en $q \leq r$ dan $p \leq r$.

Met een *maximaal* element $c \in P$ bedoelen we een element dat voldoet aan

$$(\forall x)(x \geq c \rightarrow x = c).$$

We noemen een deelverzameling $C \subset P$ een *keten* in P als $\leq$ de verzameling C lineair ordent, dat wil zeggen $p \leq q$ of $q \leq p$ voor $p, q \in C$. Ten slotte noemen we b een *bovengrens* van een keten C als $c \leq b$ voor alle $c \in C$. We zijn nu in staat om het Lemma van Zorn te formuleren.

Lemma 1 (Lemma van Zorn). *Zij $(P, \leq)$ een niet-lege partieel geordende verzameling waarin elke keten naar boven begrensd is. Dan heeft P een maximaal element ten aanzien van $\leq$.*

De Stelling van Krul

Een belangrijke stelling uit de getaltheorie is de maximale ideaalstelling van Krul, terug te vinden in [2] het college “Algebra 2 en 3” dat wordt gegeven in het tweede jaar van de bachelor Wiskunde aan de Universiteit Leiden.

Als R een commutatieve ring met eenheidselement 1 voor de multiplicatieve bewerking is en $R \neq \{0\}$, dan bezit deze ring een maximaal ideaal M . We geven het bewijs.

Stelling 2 (Stelling van Krul). *Als $R \neq \{0\}$ een commutatieve ring is met 1, dan bezit R een maximaal ideaal.*

Bewijs. Laat $R \neq \{0\}$ een commutatieve ring zijn. We bewijzen met behulp van het lemma van Zorn dat R een maximaal ideaal bezit. Beschouw de volgende verzameling idealen:

$$P = \{I \subseteq R : I \text{ is een ideaal en } I \neq R\}.$$

Dan is de inclusierelatie $\subseteq$ een *partiële ordening* op P (dit is gemakkelijk na te gaan). We moeten laten zien dat iedere keten van idealen in P een bovengrens heeft ten aanzien van de inclusierelatie. Voor de keten $\emptyset$ geldt dat het ideaal $(0) \subset R$ een bovengrens is. Laat nu $C = \{I_k\}_k$ een keten idealen zijn van R . We moeten laten zien dat er een bovengrens voor C bestaat. Bekijk $I = \bigcup_k I_k$. Het is niet moeilijk in te zien dat dit weer een ideaal oplevert. Verder geldt omdat $1 \notin I_k$ voor alle k , dat ook $1 \notin I$. Dit betekent dat $I \neq R$. Omdat $I_k \subset I$ voor alle k concluderen we dat I een bovengrens is voor C . Uit het Lemma van Zorn volgt nu dat R een maximaal ideaal bezit. $\square$

De stelling van Krul is een belangrijk resultaat in de getaltheorie die het bestaan van maximale idealen garandeert (in de aanwezigheid van het Keuzeaxioma!). Voor ringen zoals hierboven gedefinieerd is het volgende te bewijzen:

“ $I \subseteq R$ is een maximaal ideaal dan en slechts dan R/I een lichaam is.”

Hierbij is R/I het quotiëntenlichaam van R uitgedeeld naar het maximaal ideaal I . Denk bijvoorbeeld aan het lichaam $\mathbb{Z}/5\mathbb{Z}$, waarbij $\mathbb{Z}$ de ring is en $5\mathbb{Z}$ het maximale ideaal is. In het bijzonder is ieder ideaal van de vorm $p\mathbb{Z}$ waarbij p priem is een maximaal ideaal. 

Referenties

- [1] T.J. Jech. *The Axiom of Choice*. Dover Publications, 1973.
- [2] P. Stevenhagen. *Algebra 2*. Universiteit Leiden, 2010.
- [3] E. Zermelo. Beweis, dass jede Menge wohlgeordnet werden kann. *Mathematische Annalen* 59, pages 107–128, 1904.



The combined evidential value of forensic DNA profiles

Ir. J.C. de Zoete

In forensic science it is common practice to work on problems where the likelihood ratio is based on observed single piece evidence. However, in a lot of cases, there is more than one piece of evidence.

Last year, I worked on my master thesis at the Netherlands Forensic Institute. The topic I worked on was *Combined Evidential Value of Forensic DNA profiles*. During this internship, I worked on three subtopics; The combination of partial Y-chromosomal DNA profiles, the combination of autosomal and Y-chromosomal DNA profiles and the combination of low template DNA profiles. Here, I will only give a brief overview of the things I did in the first subtopic, *the combination of partial Y-chromosomal DNA profiles*. The complete thesis can be found on the website of the TU Delft library.

Forensic Statistics

The main issue in forensic science is whether a fact has occurred or not. Since in almost any case, one can never be absolutely sure whether this is the case, the important question becomes how likely it is that the fact has occurred, given the evidence. The main question for forensic statisticians is, therefore, “what is the degree of belief that hypothesis H_p is true given evidence E ?” To answer this question Bayes theorem is used.

Bayes’ Theorem

When a crime has been committed and there is a suspect, the evidence in the case is denoted with E . An explanation is needed to explain the evidence. Usually two explanations are of interest, one from the prosecutor H_p and one from the defense (the suspect) H_d (this explanation is often unknown). For instance, when E is a blood stain on the jacket of the suspect.

H_p : the suspect is the source of the blood stain.

H_d : the suspect is not the source of the blood stain.

Judges are interested in the answer to the question “which hypothesis (H_p or H_d) is more likely, given the evidence?”

Theorem 1. Bayes’ theorem

Let $\mathbb{P}(H_p)$ and $\mathbb{P}(E)$ be the probabilities that propositions H_p and E are true. Then $\mathbb{P}(H_p|E)$ is the probability that H_p is true, given that proposition E is true. If $\mathbb{P}(H_p) > 0$ and $\mathbb{P}(E) > 0$, than Bayes’ theorem states that

$$\mathbb{P}(H_p|E) = \frac{\mathbb{P}(E|H_p)\mathbb{P}(H_p)}{\mathbb{P}(E)}. \quad (1)$$

We have a formula for the conditional probability of H_p given E , the *posterior probability*. The probability $\mathbb{P}(H_p)$ is a *prior probability*, the probabilities that H_p will occur without considering the evidence. $\mathbb{P}(E|H_p)$ is the *likelihood*. Usually, a judge wants to know the ratio of posterior probabilities, $\frac{\mathbb{P}(H_p|E)}{\mathbb{P}(H_d|E)}$.

We use equation 1 to find the odds form of Bayes’ theorem,

$$\frac{\mathbb{P}(H_p|E)}{\mathbb{P}(H_d|E)} = \frac{\frac{\mathbb{P}(E|H_p)\mathbb{P}(H_p)}{\mathbb{P}(E)}}{\frac{\mathbb{P}(E|H_d)\mathbb{P}(H_d)}{\mathbb{P}(E)}} = \frac{\mathbb{P}(E|H_p)}{\mathbb{P}(E|H_d)} \cdot \frac{\mathbb{P}(H_p)}{\mathbb{P}(H_d)}. \quad (2)$$

The term $\frac{\mathbb{P}(H_p)}{\mathbb{P}(H_d)}$, the *ratio of prior probabilities*, needs to be determined by

the judge. The term $\frac{\mathbb{P}(E|H_p)}{\mathbb{P}(E|H_d)}$, the *likelihood ratio* is answered by forensic scientists. The likelihood ratio (LR) is a measure of the value of evidence regarding two hypotheses. It indicates the extent to which the evidence is in favor of one hypothesis over the other.

The combination of partial Y-chromosomal DNA profiles

Here we will discuss an approach on how to combine the evidence from several partial Y-chromosomal DNA profiles found on the same crime scene. The toy-problem which is the basis of our approach is the following:

Problem: On a t-shirt of a victim of a sex offense two DNA-traces from separate stains are secured. The traces provide two partial Y-profiles, which partly overlap. Both profiles match the DNA-profile of the suspect Teun. What is the value of the evidence of these observations?



Figure 1: The suspect Teun

In a more classical setting, i.e. a setting where there is only one crime stain, the most common hypotheses are, *the suspect is the source of the crime stain* (H_p) and *an unknown man (unrelated to the suspect) is the source of the crime stain* H_d . The evidence E can be chosen to be the match between the Y-chromosomal DNA profile of the suspect and the Y-chromosomal DNA profile obtained from the crime stain. Now the probability $\mathbb{P}(E|H_p)$ is equal to 1 (this is the probability that we find match between the DNA profile of the suspect and the DNA profile obtained from the crime stain, given that the crime stain was left by the suspect). The probability $\mathbb{P}(E|H_d)$ can be determined by estimating the rarity of the Y-chromosomal DNA profile obtained from the



crime stain using a DNA database. Suppose that this gives us an rarity of 1 in 1000, the likelihood ratio that is reported will be,

$$LR = \frac{\mathbb{P}(E|H_p)}{\mathbb{P}(E|H_d)} = \frac{1}{\frac{1}{1000}} = 1000$$

or, it is 1000 times more likely to find the evidence E when the suspect is the donor of the crime stain than when an unknown man (unrelated to the suspect) is the donor of the crime stain.

In this toy problem, we have two crime stains. Therefore, it becomes possible that the stains were left by more than one person. The hypotheses that we will use are;

- H_1 the suspect is the source of both crime stains.
- H_2 the suspect is the source of the first crime stain, some unknown unrelated man is the source of the second crime stain.
- H_3 the suspect is the source of the second crime stain, some unknown unrelated man is the source of the first crime stain.
- H_4 the suspect is not the source of the crime stains, one unknown unrelated man is the source of the crime stains.
- H_5 the suspect is not the source of the crime stains, two unknown unrelated (to the suspect and to each other) men are the sources of the crime stains.

Since we are not in a classical case where we only have two hypotheses (he did it versus someone else did it) we need to combine hypotheses. We choose to compare the hypotheses the suspect left both crime stains versus the suspect did not leave both crime stains. We combine H_2, H_3, H_4 and H_5 (which we will call H_d) and compare them with H_1 (H_p).

The evidence E is chosen to be the matches between the partial Y-chromosomal DNA profiles from the crime stains and the Y-chromosomal DNA profile of the suspect. Now we will derive the likelihood ratio for two subsets of hypotheses.

$$\mathbb{P}(E|H_d) = \frac{\mathbb{P}(E \cap H_d)}{\mathbb{P}(H_d)} = \frac{\sum_{i=2}^5 \mathbb{P}(E|H_i)\mathbb{P}(H_i)}{\sum_{i=2}^5 \mathbb{P}(H_i)}, \quad (3)$$

$$\mathbb{P}(E|H_p) = \mathbb{P}(E|H_1) \quad (4)$$

so, by combining (3) and (4),

$$\frac{\mathbb{P}(E|H_p)}{\mathbb{P}(E|H_d)} = \frac{\mathbb{P}(E|H_1) \left\{ \sum_{i=2}^5 \mathbb{P}(H_i) \right\}}{\sum_{i=2}^5 \mathbb{P}(E|H_i)\mathbb{P}(H_i)}$$

The likelihoods $\mathbb{P}(E|H_i)$ can be estimated by using a DNA database, but the prior probabilities $\mathbb{P}(H_i)$ are usually determined by the judge. In the case of two hypotheses, this can already be a hard task, but in this case, it becomes almost impossible to assign accurate or realistic prior probabilities to the hypotheses. The number of hypotheses will increase exponentially as we increase the number of crime stains. Therefore, we will determine a lower bound for the likelihood ratio, which is in favor of a suspect.

We make the following derivation:

$$\begin{aligned} \frac{\mathbb{P}(E|H_p)}{\mathbb{P}(E|H_d)} &= \frac{\mathbb{P}(E|H_1) \left\{ \sum_{i=2}^5 \mathbb{P}(H_i) \right\}}{\sum_{i=2}^5 \mathbb{P}(E|H_i)\mathbb{P}(H_i)} \\ &\geq \frac{\mathbb{P}(E|H_1)}{\max_{i \in \{2,3,4,5\}} \mathbb{P}(E|H_i)} \end{aligned} \quad (5)$$

Our result is a lower bound for the likelihood ratio where we do not need the prior probabilities of the hypotheses. However, there is something unsatisfying when using this method. The likelihood ratio is of the same order

as the likelihood ratio would have been when we compared the hypotheses that state that the suspect left **both** crime stains and the hypotheses with the largest conditional probability. When one of the alternative hypotheses state that the suspect left one of the crime stains and that an unknown unrelated person left the other, the lower bound of the likelihood ratio is equal to the likelihood ratio when we would only consider the other crime stain. In practice, the defense will most likely state that none of the crime stains originates from the suspect. Therefore another approach might be more desirable here. In practice the following pair of hypotheses might be more relevant,

H_p : the suspect left **all** crime stains.

H_d : the suspect did not leave **any** of the crime stains.

Conclusion

The number of propositions that explain the findings grows rapidly when the number of pieces of evidence increases. To evaluate the weight of the evidence given the different propositions we need to know the prior probabilities of all the propositions. We suggested a method that derives a lower bound for the likelihood ratio. This lower bound does not contain prior probabilities, which makes it easily applicable. We suggest that the best way to use it is by only comparing the hypotheses *the suspect is the source of all the evidence* versus *the suspect is not the source of any piece of evidence*. Where the latter consists of all the possible propositions (i.e. one unknown unrelated person is the source of all the evidence, one unknown unrelated person is the source of the first piece of evidence another unknown unrelated person is the source of the other pieces, ...). In a forensic report one could write,

"The probability of finding the evidence E given that the suspect left all the crime stains is at least M times more likely than when he did not leave any crime stain".

More on this subject (the combination of partial Y-chromosomal DNA profiles) and on other subjects (the combination of autosomal and Y-chromosomal DNA profiles and the combination of low template DNA profiles) can be found in my master thesis [1].

References

- [1] J.C. de Zoete, *Combined evidential value of forensic DNA profiles*, 2012.



Digitale bankroof gestopt

Erik Biemans

Erik Biemans was de afgelopen jaren een bekend gezicht op CH. Als informaticastudent en actieve CH'er heeft hij in diverse commissies gezeten, voorlichtingsdagen georganiseerd en was hij vaak aanwezig op activiteiten en in het CH-hok. Nu is hij werkzaam bij Fox-IT in Delft. Zijn werkzaamheden en ervaringen zal hij in dit interview toelichten.

Wat doe je precies bij Fox-IT?

Ik ben fraude-analist voor DetACT, een service die Fox-IT aan banken levert om fraude tegen te gaan. Jaarlijks wordt er in Nederland voor tientallen miljoenen euro's aan fraude gepleegd via internetbankieren. Voor meerdere banken monitort DetACT het internetbankieren-verkeer. Wanneer er een poging tot fraude wordt gedaan, stuurt het systeem automatisch een bericht naar de bank waarna de bank de frauduleuze transactie kan blokkeren. Het is mijn taak om dergelijke fraudegevallen te onderzoeken en het DetACT-systeem te leren hoe deze herkend kunnen worden. Ook onderzoek ik malafide software die, zonder dat de rekeninghouder dat door heeft, frauduleuze betalingen uitvoert.

Hoe ziet die fraude er dan uit?

Wij maken onderscheid tussen drie verschillende soorten fraude via internetbankieren. In de media horen we vaak over phishing, waarbij het slachtoffer (middels een site die lijkt op die van de bank of tijdens een telefoongesprek) wordt overtuigd zijn of haar gegevens af te staan. Hiermee kan een crimineel inloggen en geld overboeken. Een andere vorm van fraude wordt uitgevoerd met behulp van een trojan horse, malafide software die zich op de computer van het slachtoffer bevindt. Op het moment dat het slachtoffer naar de site van de bank gaat, zet het automatisch transacties klaar zonder dat het slachtoffer dit merkt. De derde vorm is een tussenvorm waarbij de aanval handmatig de transactie uitvoert met de gegevens die via een trojan zijn verkregen.

Hoe worden die fraudes gestopt?

DetACT geeft de bank automatisch een advies op het moment dat het systeem verdacht gedrag waarneemt. Afhankelijk van de signalen die er worden gedetecteerd wordt er een score naar de bank gestuurd. Wanneer wij bijvoorbeeld gedrag constateren dat een mens niet kan vertonen, is dit een signaal dat er iets aan de hand kan zijn; een overboeking die binnen 0,1 seconde wordt ingevoerd is hoogstwaarschijnlijk niet door een mens gedaan. Als het daarnaast ook een overboeking van een groot bedrag betreft naar een rekening in het buitenland is de kans groot dat het hier om een poging tot fraude gaat. Dit is één van de vele manieren waarop we de verschillende vormen van fraude herkennen.

Dus de banken die DetACT gebruiken hebben nooit meer last van fraude?

Helaas niet. Wanneer criminelen doorhebben dat hun aanval niet meer werkt, passen ze vaak hun methodes aan. Als de bank nieuwe fraudegevallen constateert stellen ze ons op de hoogte. Het is mijn taak om de fraudesessies te onderzoeken op unieke signalen. Wanneer ik voldoende informatie heb, zorg ik ervoor dat we DetACT bijstellen. Daarnaast krijgen wij dagelijks informatie door van InTELL, een onderdeel van Fox-IT waar de duistere hoekjes van het internet worden afgestruind op zoek naar nieuwe aanvallen. Zij infecteren zichzelf bewust met malafide software om achter de zogeheten inject code te komen, software die zich in de browser van het slachtoffer nestelt en uitgevoerd wordt wanneer de site van de bank wordt geopend. Door de inject code te analyseren kan ik een inschatting maken van de werking van de trojan, zodat een dergelijke aanval voorkomen kan worden.



Je hebt nu verteld wat je werk inhoudt, maar hoe is het om bij Fox-IT te werken?

Het burgerleven was in het begin wel wennen. Doordeweeks feesten, tot 4 uur 's nachts film kijken en uren op Facebook hangen is er nu niet meer bij maar ik vind het erg fijn om met iets zinvol bezig te zijn. Het geeft echt een kick als door mijn werk een aanval wordt tegengehouden en er ergens in de wereld een crimineel zich zit te verbijten omdat hij voor niets maandenlang bezig is geweest. De werksfeer bij Fox-IT is erg relaxed, de werkuren zijn flexibel en het is mogelijk om thuis te werken. Elke dag is er een uitgebreide lunch en iedereen kan onbeperkt frisdrank, Nespresso en versnaperingen pakken. Er worden gratis cursussen en trainingen gegeven aan werknemers, zo heb ik een uitgebreide cursus hacken gevolgd. Verder heb ik leuke collega's; er werken erg

Advertentie



FOX IT
FOR A MORE SECURE SOCIETY



DETACT
BY FOX IT



intelligente mensen die gezamenlijk een passie voor security hebben. Meerdere collega's hebben een verleden in de hack-scene en gebruiken hun kennis om zich in te zetten voor een veiligere samenleving. Voor hen is Fox leuk omdat ze hier vooral met uitdagende projecten bezig kunnen zijn.

Wat voor projecten bedoel je?

Fox-IT heeft meerdere afdelingen met allemaal hun eigen specialisatie. Wanneer een bedrijf gehackt is of er een vermoeden is van bedrijfsspionage, gaan mensen van de Forensics-afdeling op zoek naar digitale sporen. Na de hack van certificaatautoriteit DigiNotar was Fox-IT maandenlang in het nieuws, maar juist de verhalen die niet in het nieuws komen zijn het meest interessant. De Crypto-afdeling heeft onder andere een cryptografie-chip ontwikkeld voor de versleuteling van staatsgeheime informatie. Zij leveren daarnaast allerlei toepassingen aan defensie en bedrijven over de hele wereld bij wie het uitlekken van informatie gevaar voor de bevolking kan opleveren. Zelf werk ik bij de Cybercrime-afdeling, waarvan DetACT een onderdeel is. Bij Cybercrime worden penetration tests uitgevoerd bij bedrijven om na te gaan of hun systemen veilig genoeg zijn. In ongeveer 98% van de gevallen zijn de hackers van Fox-IT in staat om total ownership van de bedrijfsnetwerken van onze klanten te krijgen. Elke week worden er door Foxers presentaties gehouden over hun werk waar de meest uiteenlopende en altijd interessante verhalen worden verteld.

Hoe ben je bij Fox-IT terecht gekomen?

Ik kende Fox-IT uit het nieuws, op sites als Webwereld, Tweakers en Security.nl wordt Fox vaak genoemd. In het voorprogramma van de studiereis in 2009 naar Amerika zijn we bij Fox-IT langsgesegaan waar we onder andere een hackdemonstratie kregen en de analyse van een stukje malware. Ik vond het erg interessant maar de snelheid waarmee er door de assembly-code van de 0-day pdf exploit werd gesprongen gaf me het gevoel dat ik de kennis miste om bij Fox-IT te kunnen werken.

Maar je zit er nu toch.

Vorig jaar was ik op zoek naar een bachelorproject en op advies van een paar CH'ers zijn mijn drie groepsgenoten en ik bij Fox-IT gaan kijken. Ze hadden een interessante opdracht bij de Crypto-afdeling van Fox-IT. We hebben een filtering framework voor de DataDiode gebouwd waarmee selectief eenrichtings-netwerkverkeer kan worden gefilterd. Onze begeleider was hier tevreden mee en heeft ons alle vier een baan aangeboden om de software verder te ontwikkelen. Hoewel ik het project erg interessant vond, sprak de Cybercrime-afdeling mij meer aan. In een gesprek met Walter Doorduyn is gekeken of er ook andere mogelijkheden binnen Fox-IT waren, waardoor ik bij DetACT terecht ben gekomen. Het feit dat ik geen master heb gedaan was geen enkel probleem, er wordt meer gekeken naar de motivatie en de vaardigheden van iemand dan naar de diploma's of certificeringen.

Wat kan iemand bij Fox-IT doen?

Studenten die net als ik een bachelorproject willen doen zou ik zeker aanraden langs te komen. Er zijn momenteel meerdere projecten waar info's en wisk'o's voor gezocht worden en er zijn diverse afstudeeropdrachten beschikbaar. Uiteraard is het ook mogelijk om te solliciteren, we zijn hard op zoek naar fulltime developers maar er zijn ook parttime functies beschikbaar voor mensen die naast hun studie willen werken.

Zijn er nog dingen om rekening mee te houden?

Voordat iemand bij Fox-IT gaat solliciteren of stagelopen is het belangrijk dat hij of zij beseft dat het een bedrijf is waar veel gevoelige informatie wordt gedeeld. Voordat je aan de slag kunt moet je eerst een screening door een onderzoeksbureau en de AIVD ondergaan om te kijken of je betrouwbaar en niet chantabel bent. Tijdens het werk kun je met je collega's bijna alles bespreken maar als je in de /Pub van een biertje aan het genieten bent zul je de dingen die je meemaakt geheim moeten houden. Omdat de meeste klanten anoniem willen blijven wordt vaak gebruik gemaakt van codenamen, werknemers worden getraind om continu op hun hoede te zijn voor spionagepraktijken. Zelf vind ik dit juist spannend maar ik kan me voorstellen dat het niet voor iedereen is weggelegd.

Wat moet iemand doen voor een stage of baan?

Ik heb zelf op de website gekeken naar vacatures maar eerlijk gezegd schrokken de eisen mij een beetje af. Er is wel veel te vinden over de verschillende projecten die er zijn dus het is zeker interessant om even rond te klikken. Als je meer wilt weten over mijn baan of werken bij Fox-IT in het algemeen kun je mij mailen (erik.biemans@fox-it.com) of een bericht sturen via Facebook of LinkedIn. Als je denkt dat Fox-IT jou iets te bieden heeft, kun je ook contact met Walter Doorduyn opnemen (doorduyn@fox-it.com). Walter heeft er voor gezorgd dat ik nu bij Fox-IT werk en hij kan samen met jou een interessant bachelor- of afstudeerproject uitzoeken of bekijken welke banen jou aanspreken.





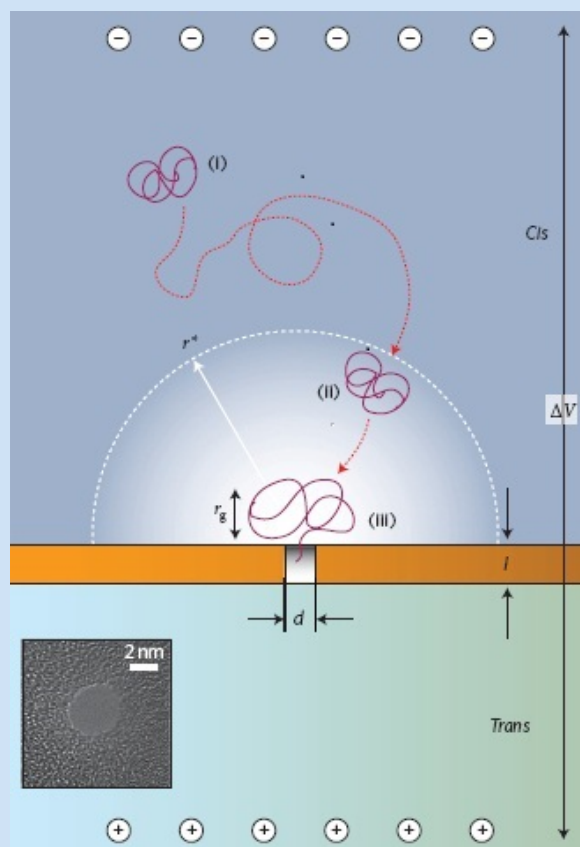
Modelleren in de nanofysica

Johan Dubbeldam

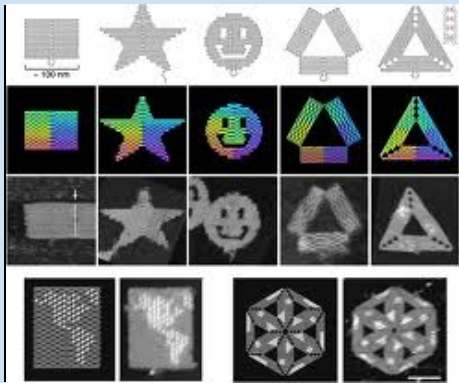
Het uitlezen van een menselijk genoom is een fantastische prestatie die voor het eerst in 2003 werd verwezenlijkt. Het project om het eerste menselijke genoom, dat tussen de 20000 en 25000 genen bevat, te ontrafelen kostte 15 jaar en een honderden miljoenen euros. Sindsdien zijn er enorme ontwikkelingen geweest en lijkt een toekomst waarin iedereen zijn genoom kan laten bepalen erg dichtbij. Men kan zich indenken dat deze ontwikkeling in de toekomst ook grote maatschappelijke gevolgen zal hebben. Hier zal ik me echter bezighouden met de rol van wiskunde in het uitlezen van het menselijk DNA. Het blijkt dat er bij het correct uitlezen van een genoom een aantal verschillende wiskundige disciplines van belang zijn.

Het ontrafelen van DNA, waarin de erfelijke code van een organisme is opgeslagen is een moeilijke en tijdrovende klus. Aanvankelijk werd DNA in stukjes geknipt die dan werden geanalyseerd, waarna de oorspronkelijke volgorde van de nucleotiden weer moest gereconstrueerd. Deze zgn Sanger methode heeft echter het nadeel dat er fouten kunnen optreden in de reconstructie van de oorspronkelijk volgorde en dat hij erg langzaam is. Sinds in 1996 door Kasianowicz voor het eerst werd aangetoond dat je grote strengen DNA op een lineaire manier door een smalle porie in een membraan kan transporteren, zogenaamde *translocatie*, en daarbij verschillende nucleotiden kan onderscheiden heeft het onderzoek naar dit fenomeen een grote vlucht genomen, omdat het suggereert dat het mogelijk is om tenminste 1000 maal langere ketens dan in de Sanger methode uit te lezen. Het ultieme doel was het realiseren van een apparaat dat van ieder mens zijn DNA kan aflezen in redelijke tijd en voor geringe kosten (< 1000 euro). Met het verschijnen van de aankondiging van het bedrijf *Oxford Nanopores* dat zij een device op de markt gaan brengen dat dit mogelijk maakt lijkt dit doel daadwerkelijk gerealiseerd te worden.

Dit betekent echter niet het einde van de bedrijvigheid op het gebied van nanogaatjes, zoals ik dadelijk zal laten zien. Eerst zal ik echter kort het translocatie proces schetsen, waarbij een lange keten door een nanogaatje wordt getransporteerd. We stellen ons een polymeerketen (bijv. DNA) voor als een verzameling kralen, monomeren genaamd, die onderling verbonden zijn door veertjes. Deze keten zit initieel aan één kant (cis-zijde) van het membraan. Doordat de keten geladen is en er een potentiaalverschil over het membraan wordt aangebracht en DNA negatief geladen is, zal het DNA worden aangetrokken door de positieve ionen aan de trans-zijde van het membraan. De lange keten zal zich eerst bewegen totdat hij in de buurt van de porie is. Dit noemt men de *vangfase*. Deze vangfase heeft een interessante dynamica die tot op heden nauwelijks is onderzocht; zie ook de figuur boven aan de bladzijde



Figuur 1: DNA wordt eerst gevangen en vindt dan zijn weg door de porie.



Figuur 2: Met DNA kun je zelfs origami spelen.

waar de vangfase duidelijk is afgebeeld. Als de keten voldoende dicht in de buurt van de porie is gekomen, wurmt de keten een van zijn uiteinden door het gaatje heen, waarna het translocatieproces van start gaat.

De wiskundige beschrijving van translocatie is een modelleerprobleem, dat ook kennis van de fysica van niet-evenwichtsprocessen vereist. Omdat de beweging van elk der monomeren te ingewikkeld is, wordt meestal alleen een beweging-vergelijking opgesteld voor de ketenlengte die aan een enkele zijde van het membraan verblijft. De evolutie van deze lengte voldoet aan een stochastische differentiaalvergelijking, omdat elk der monomeren door botsingen met vloeistofmoleculen voortdurend een andere snelheid krijgt.

Tot dusver is het ons niet gelukt een microscopische afleiding te vinden voor deze differentiaalvergelijking. Wel hebben we door middel van numerieke simulaties kunnen aantonen dat evolutie van de ketenlengte aan de transzijde een niet-Markov proces is. Dit wil zeggen dat de evolutie van lengte aan de transzijde afhangt van de geschiedenis en niet slechts van de vorige tijdstap. In het bijzonder blijkt dat een stap naar rechts van de keten met grote kans wordt gevolgd door een stap naar links. De snelheid van de monomeer in de porie is dus anti-gecorrleerd in de tijd.

Dit soort processen zijn eerder door Mandelbrot, bekend van de fractalen, en Ness bestudeerd in de zestiger jaren en vormen de klasse van zogenaamde fractionele Brownse beweging. Fractionele Brownse beweging is een continu en Gaussisch proces waarbij de variantie in de tijd toeneemt als t^{2H} , waarbij H de Hurst exponent is. Het is een directe generalisatie van gewone Brownse beweging tot processen waarbij de incrementen niet meer onafhankelijk zijn. Preciezer geformuleerd wordt de covariantie gegeven als $Cov(s(t)s(\tau)) = (|t|^{2H} + |\tau|^{2H} + |t - \tau|^{2H})$ Als $H = \frac{1}{2}$ dan neemt de variantie lineair toe in de tijd en is er sprake van gewone diffusie. Als $H < \frac{1}{2}$ dan is het proces subdiffusief en heeft het proces een negatieve auto-correlatie. Deze fractionele processen zijn overigens ook relevant in de financiële wetenschap. Het mag dus duidelijk zijn dat het translocatieprobleem een schat aan wiskundig interessante problemen herbergt, zowel op het gebied van de modellering als op het gebied van de kansrekening/stochastische processen.

Het onderzoek naar nanogaatjes en translocaties is, zoals al eerder genoemd, nauw verweven met technologie die ons leven in de toekomst weleens flink zou kunnen veranderen.

Voor de nanogaatjes ziet de toekomst er rooskleurig uit. Er zijn tal van nieuwe toepassingen waarin nanoporiën een belangrijke rol spelen. Het is tegenwoordig zelfs mogelijk om een nanogaatje te bekleden met een DNA coating, dit heet DNA origami. Het DNA is dan eerst in zo'n vorm gevouwen dat wanneer dat in een nanopore wordt ingebracht deze pore specifieke gewenste eigenschappen geeft. Dit zou bijvoorbeeld kunnen worden toegepast om snel het RNA van een specifiek virus te detecteren. Aan de TU wordt dit soort experimenteel onderzoek ook verricht. De groep moleculaire biofysica die door Cees Dekker geleid wordt en waarmee wij ook samenwerken, is een van de leidende laboratoria in de wereld op het gebied van experimenten met solid state nanoporiën. Het is indrukwekkend om te zien hoe met krachten van 10pN met behulp van zgn. magnetische tweezer een DNA keten door een nanogaatje kan worden getrokken. Ook een wiskundige kan deze mooie technologie gegrepen worden.

Ik hoop dat ik in dit korte stuk duidelijk heb kunnen maken dat polymeer translocatie een fascinerend fenomeen is waar veel wiskunde aan te pas komt. Studenten die mee willen denken over het construeren van mathematisch-fysische modellen voor DNA translocatie, leuke toepassingen zoeken voor fractionele Brownse beweging of geïnteresseerd zijn in numerieke simulaties van DNA moleculen kunnen voor meer informatie altijd langskomen.



De verblijftijd van water in een meer met behulp van een deeltjesmodel

Margriet Nieuwenhuis

Hieronder volgt een beknopte beschrijving van de inhoud van mijn bachelor eindproject. Het project betrof het maken van een deeltjesmodel voor de verblijftijd van water in een meer.

De verblijftijd van water in een meer wordt ook wel de leeftijd van water in een meer genoemd. Hoe lang bevindt het water zich in het meer? De verblijftijd speelt een rol als er wordt gekeken naar de ontwikkeling van gewassen en de verspreiding van schadelijke stoffen. De beweging van deeltjes in water wordt gegeven door de advection-diffusie vergelijking. Advection geeft aan dat het deeltje in het water wordt meegenomen door de stroomrichting van het meer. Diffusie is de willekeurige beweging van een deeltje zelf. Elk deeltje beweegt onafhankelijk van andere deeltjes.

De advection-diffusie vergelijking is op twee manieren te benaderen om te rekenen aan de verblijftijd van water. De eerste manier beschrijft de concentratie van water met behulp van partiële differentiaalvergelijkingen. De oplossing van die vergelijkingen worden gesimuleerd met een numeriek model. De tweede manier is door de beweging van elk deeltje apart te bekijken in het model. De eerstgenoemde manier was eerder onderwerp van een bachelorproject. Daarom heb ik onderzoek gedaan naar de leeftijd van water met behulp van een deeltjesmodel. Doel van het project is om na te gaan of het mogelijk is om een deeltjesmodel op te stellen voor de leeftijd van water. Daarnaast is er gekeken naar de leeftijd van de deeltjes in het meer en de invloed van de contouren van het meer op de leeftijd van het water.

Advection-diffusie vergelijking

De twee-dimensionale advection-diffusie vergelijking luidt als volgt:

$$\frac{\partial HC}{\partial t} = -\frac{\partial uHC}{\partial x} - \frac{\partial vHC}{\partial y} + \frac{\partial^2 HD\partial C}{\partial^2 x} + \frac{\partial^2 HD\partial C}{\partial^2 y}$$

De eerste twee termen van de rechterkant van de vergelijking geven de advection aan en de laatste twee termen de diffusie. De termen in de vergelijking hebben de volgende betekenis:

- $C(x, y, t)$: de concentratie van een zekere stof in het water;
- $H(x, y, t)$: de waterdiepte;
- $u(x, y, t)$: de snelheid van het water in de x-richting;
- $v(x, y, t)$: de snelheid van het water in de y-richting;
- $D(x, y)$: de dispersie coëfficiënt.

Constructie deeltjesmodel

Hieronder wordt het deeltjesmodel voor de leeftijd afgeleid.

Partiële differentiaalvergelijkingen voor de leeftijd

De beweging van een waterdeeltje wordt beschreven met behulp van een stochastische differentiaalvergelijking.

$$\frac{dX_t}{dt} = f(X_t, t) + g(X_t, t)N_t \quad (1)$$
$$X(t_0) = X_0$$

Deze vergelijking bevat een stochastisch proces X_t als variabele. De N_t -term is een continu Gaussisch witte ruis (stochastisch) proces zodanig dat (1) Markov is. Een stochastisch proces is het Wiener proces W_t . De eigenschappen van zo'n Wiener-proces zijn als volgt:

- De beginconditie is nul, $W_0 = 0$;
- Voor $0 \leq s < t \leq T$ is de toename $W_t - W_s$ een Gaussisch random variabele met verwachting nul en variantie $t - s$, $W_t - W_s \sim N(0, t - s)$.

Het blijkt dat de afgeleide van een Wiener-proces voldoet aan de eisen van een continu Gaussisch witte ruis proces. Dit kan op de volgende manier herschreven worden:

$$W'_t = \frac{dW_t}{dt} = N_t \iff dW_t = N_t dt$$

Het is mogelijk om deze term te substitueren in (1). Hiermee kan een stochastische differentiaalvergelijking worden beschreven met een Wiener proces als ruis-input.

$$\begin{cases} dX_t = f(X_t, t)dt + g(X_t, t)dW_t \\ X(t_0) = X_0 \end{cases} \quad (2)$$

Eenmaal integreren van (2) en invulling van de beginvoorwaarde, levert:

$$X_t = X_{t_0} + \int_{t_0}^t f(X_s, s)ds + \int_{t_0}^t g(X_s, s)dW_s \quad (3)$$

Voor de berekening van de laatste integraal in (3), het integreren naar een Wiener-proces, wordt een speciale integraal gebruikt: een Ito-integraal.

Deeltjesmodel met leeftijd

Nu is de oplossing van een stochastische differentiaalvergelijking in (3) volledig gedefinieerd. De volgende stap is om een numeriek schema te gebruiken [1]. Neem Δt als een kleine tijdstap:

$$\begin{aligned} X_{t+\Delta t} &= X_t + \int_t^{t+\Delta t} f(X_s, s)ds + \int_t^{t+\Delta t} g(X_s, s)dW_s \approx \\ &\approx X_t + \int_t^{t+\Delta t} f(X_t, t)ds + \int_t^{t+\Delta t} g(X_t, t)dW_s = \\ &= X_t + f(X_t, t)\Delta t + g(X_t, t)(W_{t+\Delta t} - W_t) \end{aligned} \quad (4)$$

Dit schema heet het Euler-schema. In de laatste uitdrukking zijn alle termen bekend, mits X_t bekend is. Merk op dat we ook bekend zijn met het verschil van twee Wienerprocessen: dit is normaal verdeeld met verwachting nul en variantie $(t + \Delta t) - t = \Delta t$ en er geldt $dW_t = N_t dt$. Hiermee kan (4) worden herschreven tot:

$$X_{t+\Delta t} = X_t + f(X_t, t)\Delta t + g(X_t, t)N_t \Delta t \quad (5)$$



Fokker-Planck vergelijking

Met behulp van de Fokker-Planck vergelijking is het mogelijk om meer informatie te winnen over de kansdichtheidsfunctie van X_t . De Fokker-Planck vergelijking luidt als volgt:

$$\frac{\partial p}{\partial t} = - \sum_{i=1}^n \frac{\partial f_i p}{\partial x_i} + \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^m \frac{\partial^2 (g g^T)_{ij} p}{\partial x_i \partial x_j}, \quad t \geq t_0 \quad (6)$$

Hierbij is $p(x, t)$ de kansdichtheidsfunctie van de random variabele X_t uit (2), f_i en g_i de i -de component van respectievelijk de vectoren f en g van lengte n uit (2). In ons geval zijn de vectoren f en g van lengte 2 uitdrukkingen voor de x - en y -richting.

Het is mogelijk om de advection diffusie vergelijking te koppelen aan een deeltjesmodel met behulp van de Fokker-Planck vergelijking door de substitutie $HC = p$.

$$\frac{\partial p}{\partial t} = - \frac{\partial(u + \frac{\partial H D}{\partial x})p}{\partial x} - \frac{\partial(v + \frac{\partial H D}{\partial y})p}{\partial y} + \frac{\partial^2 D p}{\partial^2 x} + \frac{\partial^2 D p}{\partial^2 y}, \quad t \geq t_0$$

Hieruit worden de volgende functies voor de beweging in de x - en y -richting gevonden:

$$\begin{aligned} f_1(x, y) &= u + \frac{\partial H D}{\partial x} & f_2(x, y) &= v + \frac{\partial H D}{\partial y} \\ g_1(x, y) &= \sqrt{2D} & g_2(x, y) &= \sqrt{2D} \end{aligned}$$

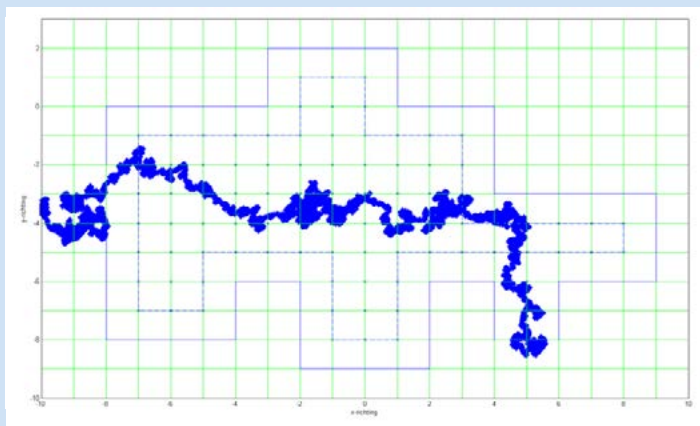
Hieruit volgt het onderliggende deeltjesmodel:

$$\begin{aligned} dX_t &= (u + \frac{\partial H D}{\partial x})dt + \sqrt{2D}dW_t^x \\ dY_t &= (v + \frac{\partial H D}{\partial y})dt + \sqrt{2D}dW_t^y \end{aligned} \quad (7)$$

Elk deeltje heeft een beweging in de x - en y -richting. De verplaatsing die wordt veroorzaakt door advection, bestaat uit de beweging van het water (in de x - of y -richting) en uit de beweging veroorzaakt door de diffusie coëfficiënt D . De verplaatsing die wordt veroorzaakt door diffusie, wordt bepaald door een random-term.

Beweging in het meer

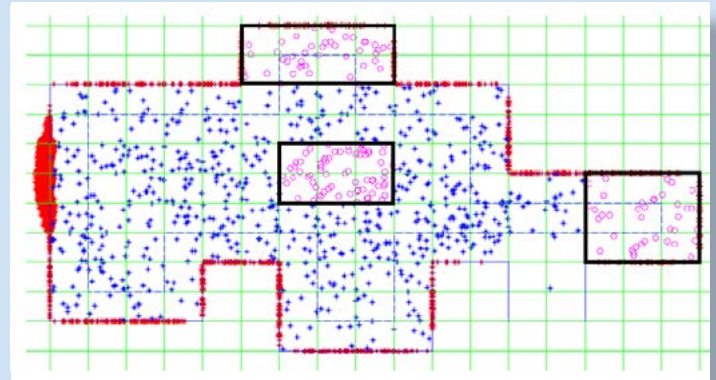
Met het afgeleide deeltjesmodel is het mogelijk om simulaties te doen. De simulaties zijn uitgevoerd in een zelfbedacht meer. Het meer bevat twee openingen: het water komt rechts onder het meer in en verlaat links in het midden het meer. De beweging van een deeltje is te zien in figuur 1.



Figuur 1: Beweging van een deeltje

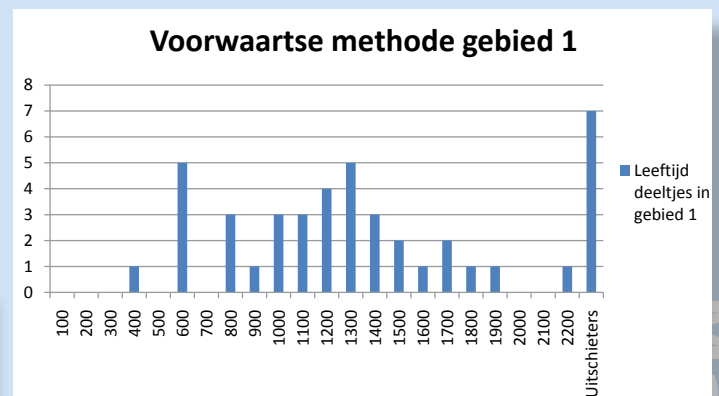
Daarna zijn er simulaties uitgevoerd waarbij er in totaal 11000 deeltjes het meer in worden gelaten, 550 deeltjes per 5 seconden. Elk deeltje heeft een variabele die bijhoudt wat de leeftijd in het meer is. Om te kijken wat de invloed

van de contouren van het meer zijn, zijn er drie deelgebieden aangebracht op verschillende locaties in het meer. Een deelgebied bevindt zich direct rechts van de inlaat van het meer, het tweede deelgebied in het midden van het meer en het derde deelgebied daar iets boven.



Figuur 2: Simulatie met de deelgebieden

In figuur 2 is de situatie in het meer aan het einde van de simulatie te zien. In de figuur zijn verschillende kleuren en vormen deeltjes te onderscheiden. De rode sterretjes geven de deeltjes aan die aan het einde van de simulatie het meer hebben verlaten. Dit kan via de 'normale' uitgang: de opening aan de linkerkant van het meer. Echter, doordat het een model is zijn er deeltjes die via de niet-reguliere manier het meer verlaten. De blauwe sterretjes zijn de deeltjes die zich aan het eind van simulatie in het meer bevinden. De roze rondjes voldoen hier ook aan, maar bevinden zich in een van de deelgebieden. Als voorbeeld laat figuur 3 de leeftijd van de deeltjes in deelgebied 1 zien.



Figuur 3: Verdeling leeftijd deeltjes deelgebied 1

De gemiddelde verblijftijd in het meer is 400 en uit de simulatie blijkt dat er in deelgebied 1 veel 'oudere' deeltjes zijn.

Conclusie

Doel van het project is om te onderzoeken of het mogelijk is de verblijftijd van water te bepalen met behulp van een deeltjesmodel. Zoals hierboven beschreven is, is dit inderdaad mogelijk. Daarnaast is uit de simulaties gebleken dat inhammen aan de rand van het meer de verblijftijd van deeltjes in het meer aanzienlijk kunnen laten toenemen.

Referenties

- [1] P.E. Kloeden and E. Platen, *The numerical solution of stochastic differential equations*, 1992, Applications of mathematics series, 23

Gaaan!

HET TALENT- EN AMBITIEPLATFORM VAN KPMG

DE AVOND VOOR HET AFRONDEN VAN
HET FUSIERAPPORT VOOR EEN
INTERNATIONALE BIERBROUWER



Je kunt als bedrijf nog zo veel willen, je krijgt pas wat voor elkaar met goede mensen.

Over KPMG

KPMG biedt vanuit zijn adviespraktijk (Audit) en accountancypraktijk (Advisory) diensten aan uiteenlopende organisaties. Van multinationals tot de nationale- of regionale 'middenmarkt'. We hebben 135.000 medewerkers wereldwijd. Alleen al in Nederland zijn er 4.000 actief, verspreid over 15 kantoren. We zijn een ambitieus bedrijf: we willen nummer één zijn in ons vak. Op veel gebieden is dit al zo. Onze adviseurs binnen de fusie- en overnamepraktijk zijn dagelijks betrokken bij veruit de meeste transacties in Nederland. En onze accountants controleren de meeste aan de AEX- en MIDCAP-beurs genoteerde ondernemingen in Nederland. Maar we willen ook nummer één zijn in de markten en op de terreinen waar we dat nog niet zijn.

...over KPMG Advisory

Onder de noemer KPMG Advisory leveren we tactisch en strategisch advies op diverse terreinen aan de meest uiteenlopende organisaties. Van grote (inter) nationale ondernemingen tot non-profitorganisaties en overheidsinstellingen. We adviseren op een breed terrein. Denk aan organisatieadvies, IT-advies, restructuring, valuations en advies bij fusies en overnames. Onze opdrachtgevers waarderen ons met name vanwege onze teamgeest en onze grote kennis van zaken.

Gááán!

We zijn voortdurend op zoek naar talent. Naar mensen met dezelfde passie. Die gedreven en betrokken zijn en het beste uit zichzelf willen halen door heel goed samen te werken. Zit jij zo in elkaar en ben je daarnaast analytisch sterk? Dan wordt het tijd om kennis te maken. Dat kan via www.gaaan.nu,

ons talent- en ambitieplatform. Gááán! helpt je bij je studie en je carrière en stelt je in staat contacten te leggen met KPMG'ers. Maar je vindt er ook meer informatie over jouw carrièremogelijkheden als adviseur.

Aan de slag als adviseur

Bij KPMG Advisory begin je als junior adviseur en start je direct met uitdagende adviesopdrachten voor verschillende organisaties. Daarbij lever je op diverse gebieden een bijdrage aan rapportages, presentaties en adviezen. Je ontwikkeling wordt aangevuld met doelgerichte opleidingen. Afhankelijk van je universitaire studie en interesse kun je kiezen uit de verschillende onderdelen binnen Advisory. Van organisatieadvies tot fusies en overnames, van het kwantificeren van complexe financiële- en operationele risico's tot tactisch en strategisch IT-advies. De keuze is aan jou.

Waar je ook voor gaat: kansen genoeg om samen met je collega's aan iets moois te bouwen. Wij zouden zeggen: Gááán!

Meer informatie

Ga naar www.gaaan.nu of maak een afspraak met het KPMG Recruitment Centre (020) 656 7162 of mail naar recruitment@kpmg.nl.

KPMG Recruitment Centre

Laan van Langerhuize 1
1186 DS Amstelveen
(020) 656 7162
recruitment@kpmg.nl / www.gaaan.nu



Een jonge onderzoeker

Sonja Cox

Mijn promotor Jan van Neerven gaf mij bij mijn promotie een zilveren

uiltje. Niet omdat ik bekend sta als dichtgetikt lid van Christiaan

Huygens, maar omdat ik – naar zijn zeggen – wellicht wat wijsheid kan gebruiken.

Want wat is wijsheid? Tijdens mijn afstuderen bij de vakgroep functionaalanalyse in Delft besloot ik dat ik absoluut niet bij die ‘verschrikkelijk theoretische’ vakgroep wilde promoveren (als ware Delftse student vond ik functionaalanalyse het summum van abstracte wiskunde...) Eigenlijk wilde ik helemaal niet promoveren, ik wilde na mijn afstuderen eindelijk eens wat nuttigs doen.

Met mijn bul op zak heb ik mezelf tien weken lang buitengewoon nuttig gemaakt met koken voor mijn huisgenoten en nadenken over de zin van het leven. Vervolgens ben ik begonnen aan een promotie bij de vakgroep statistiek in Leiden. Statistiek leek me namelijk wel nuttig. Helaas zag ik mijn toenmalige promotor amper. Hij had zaken aan zijn hoofd die – objectief gezien – belangrijker waren dan een promovenda. Ik was uiteraard niet in staat dit objectief te zien, en tot overmaat van ramp ontdekte ik bij de eerste conferentie waar ik heen gestuurd werd dat ik alleen de theoretische voordrachten leuk vond.

Kortom, een half jaar na mijn afstuderen klopte ik bij Jan van Neerven en Ben de Pagter aan, met het verzoek alsnog bij functionaalanalyse in Delft te mogen promoveren. Het mocht. Ik begon mijn promotie bij Birgit Jacob, en toen zij na negen maanden een hoogleraarspositie in Paderborn kreeg aangeboden, verklaarde Jan zich bereid mij alsnog onder zijn hoede te nemen.

Drie jaar later begon ik weer te jengelen. Ik zei na mijn promotie niet aan de universiteit te willen blijven, want dan moet je eigenlijk een paar jaar naar het buitenland, en dat is maar lastig. Bovendien, iedereen wil hoogleraar worden dus dat lukt toch niet. En sowieso, moest ik niet eens de ‘echte’ wereld in? Nee, een postdoc positie wilde ik absoluut niet.

Je begrijpt het al. Momenteel ben ik postdoc aan de ETH Zürich. Ik hoop dat het zilveren uiltje binnenkort zijn werk gaat doen.

Een jonge onderzoeker op reis

Een paar weken geleden kreeg ik het verzoek een bijdrage te leveren aan het MaCHazine, verpakt in de woorden: “Als succesvolle, vrouwelijke en jonge onderzoeker bent u een voorbeeld voor de vele studenten.” Gezien wat ik hierboven beschreef, moest ik toen even in mijn arm knijpen.

Vervolgens heb ik even overwogen een ronkend succesverhaal te schrijven. Dat zou immers passen bij het instituut waar ik nu zit. Ik zal dat even uitleggen voor degenen die niet bekend zijn met de ETH Zürich.

In Zwitserland is er een gigantisch grote berg geld en er zijn twee federale universiteiten: de ETH Zürich en de EPF Lausanne. De gigantisch grote berg geld wordt over die twee universiteiten verdeeld, met als gevolg deze universiteiten, en met name de ETH Zürich, geen last hebben van valse bescheidenheid. Betreffende het werven van hoogleraren zegt mijn afdelingshoofd bijvoorbeeld: “Als wij een aanbod doen, dan wordt het aangenomen. Punt.” En betreffende de beschikbaarheid van boeken in de bibliotheek: “gibt es nicht” gibt es nicht an der ETH”. Om een idee te geven van de sfeer heb ik een foto meegestuurd (red. linksboven) van de ingang van het hoofdgebouw.

Wat mij betreft: ik heb de eer en het genoegen twee jaar aan dit formidabele instituut te mogen werken (en wel op de zolder van het hoofdgebouw). Het mooie van hier werken is dat er veel geweldige wiskundigen langskomen. Maar het nadeel van hier werken is ... dat er veel geweldige wiskundigen langskomen. Het aanbod is zo enorm, en de mensen hebben het zo druk, dat men weinig tijd voor elkaar heeft – in ieder geval in mijn vakgroep. Zo gaan de hoogleraren in het algemeen niet met hun promovendi en postdocs lunchen, hetgeen ik jammer vind.

De reden dat er van beginnende academici verwacht wordt dat ze wat tijd aan verschillende universiteiten doorbrengen is natuurlijk dat we dit soort verschillen observeren – en natuurlijk dat we een netwerk opbouwen. Naast de ETH heb ik ook een paar maanden aan UNSW (Sydney) en de universiteit van Innsbruck mogen snuffelen, en inmiddels vrees ik dat ik binnenkort ga piepen dat ik juist niet naar Nederland wil, want dat is maar saai.

Mijn onderzoek

Aangezien ik bijna vijf jaar lang mijn ziel en zaligheid erin heb gelegd, moet ik kort uitleggen waar mijn promotieonderzoek over ging. Het grootste deel van het werk ging over het benaderen van oplossingen van stochastische partiële differentiaalvergelijkingen. Een partiële differentiaalvergelijking is, zoals jullie allemaal weten uit het eerste jaar of van wikipedia, een vergelijking die de partiële afgeleiden van een onbekende functie van twee of meer variabelen bevat. Mijn favoriete voorbeeld is de inhomogene warmtevergelijking in één dimensie:

$$\frac{\partial}{\partial t} u(t, x) = \frac{\partial^2}{\partial x^2} u(t, x) + f(t, x), \quad (t, x) \in (0, \infty) \times (0, 1).$$

Formule 1: Een inhomogene warmte vergelijking in één dimensie.

Hier is t de tijdparameter en x de ruimteparameter, en de (onbekende) $u(t, x)$ geeft de temperatuur op tijdstip t en locatie x weer. De functie f is gegeven, dit is een functie die de externe toevoer van warmte weergeeft.

Als de functie f in (1) niet deterministisch is, maar stochastisch, dan spreekt men van een stochastische partiële differentiaalvergelijking. Het moge duidelijk zijn dat $u(t, x)$ in dat geval een toevalsvariabele is voor iedere t en x .

In het algemeen is het niet mogelijk een expliciete oplossing te vinden voor een stochastische partiële differentiaalvergelijking. Om inzicht te krijgen in het gedrag van de oplossing, moet je dus numerieke simulaties doen. Aangezien de oplossing een toevalsvariabele is, brengt dit extra moeilijkheden met



zich mee: je moet bijvoorbeeld aangeven in welke zin de numerieke benadering convergeert: bijna zeker, in moment, of misschien alleen maar zwak? Vaak krijg je verschillende convergentiesnelheden afhankelijk van welke vorm van convergentie je beschouwt. Ik hou mij bezig met de vraag of een numerieke oplossing convergeert, en zo ja, in welke zin, en hoe snel.



Daarnaast vind ik het ook leuk om na te denken over hoe sommen van vectorwaardige stochastische processen zich gedragen als ze niet helemaal, maar wel een beetje onafhankelijk van elkaar zijn. Dit heeft te maken met het kunnen definiëren van de stochastische integraal van

een vector-waardig proces – en dat is essentieel om de oplossing van een stochastische partiële differentiaalvergelijking te kunnen definiëren.

Mijn studententijd

Het is verdomd lang geleden. Echt. Toen ik begon met studeren kon je met gulden betalen en stond een jaar studeren gelijk aan 42 studiepunten – en ik haalde in mijn eerste jaar 63 studiepunten. De voor de hand liggende conclusie is dat ik geen feestbeest was. Die conclusie is terecht, al is het in latere jaren wel een beetje goed gekomen.

Overigens zat ik wel – oh ironie! – in de CH-feestcommissie (zie foto). Verder heb ik ooit een jaarboek helpen maken, ik ben mee geweest op studiereis, en ik heb gepoogd mijn studiegenoten te vertegenwoordigen in de opleidingscommissie wiskunde. Ik woonde in verschillende studentenhuizen, hetgeen ik iedereen kan aanraden, al was het maar omdat alleen wonen dan weer zijn charme krijgt.

Ik werd in het eerste jaar lid van de Koornbeurs, maar het duurde enkele jaren voordat ik doorhad dat dat best een gave vereniging is – zoiets duurt wat langer als je 't niet tijdens een KMT door je strot gedouwd krijgt.



Ik had niet meer willen feesten, maar achteraf had ik een tijd naar het buitenland gewild tijdens mijn studie – toentertijd dacht ik dat je dan alleen nog meer moest feesten, maar je schijnt ook iets te kunnen leren in het buitenland.

Promoveren?

Ter afsluiting nog een klein stappenplan mocht je overwegen te promoveren.

Stap 1. Bedenk of je onderzoek doen leuk vindt. De volgende vragenlijst helpt je daarbij (als je een wiskundestudent bent).

1. Vind je je afstudeerproject leuk?
2. Heb je weleens gedroomd dat je iets kon bewijzen? (En kon je 't ook toen je wakker werd?)
3. Heb je weleens na middernacht op een feestje geprobeerd uit te leggen dat een stochastisch gewoon een meetbare afbeelding op een maatruimte met maat één is? Of dat er een topologisch bewijs bestaat dat er oneindig veel priemgetallen zijn? Of wat precies de hypothese $P = NP$ inhoudt?

Als je op alle vragen 'ja' hebt geantwoord, dan vind je onderzoek vermoedelijk leuk – ook als je op sommige feestjes domweg veel gezopen hebt en je dromen meestal over... eh, vliegen ofzo gaan.

Als je op géén van de vragen 'ja' hebt geantwoord, kan het nog steeds zijn dat je onderzoek leuk vindt, al acht ik de kans minder groot. Je kunt een promotie zien als een mogelijkheid om erachter te komen.

In ieder geval is het feit dat je onderzoek denkt leuk te vinden waarschijnlijk de enige goede reden om aan een promotie te beginnen.

Stap 2. Zoek een goede promotor. Een goede promotor is iemand die toonaangevend onderzoek doet in een richting die jij interessant vindt, redelijk veel tijd voor je heeft, je voldoende vrijheid geeft, en – last but not least – iemand waarmee je op persoonlijk vlak goed kan opschieten. Ik denk dat een promotor die je al redelijk goed kent te prefereren is boven een promotor aan een geweldig instituut ergens ver weg, maar ik ben nogal risicomijdend. Overigens kun je meestal een deel van je promotie aan een ander instituut doen (lees: je kunt heus wel een tijd naar het buitenland als je dat wilt).

Stap 3. Vergeet niet dat je 't leuk vindt. Dit is de moeilijkste stap. Op het eerste gezicht lijkt promoveren misschien geweldig: het is een soort betaald studeren, je kunt fijn reizen naar conferenties, leren van vooraanstaande onderzoekers op zogenaamde summer- en winter schools, en niemand verlangt van je dat je verantwoording aflegt... Aye, there's the rub – want je zult bij jezelf verantwoording moeten afleggen. Het allermoeilijkste van promoveren is het accepteren dat je proefschrift bij lange na niet volmaakt is, dat de resultaten die je hebt bewezen lang niet zo spectaculair zijn als je had gehoopt, en dat er op de hele wereld maar één persoon is die het proefschrift van begin tot einde heeft gelezen: jijzelf. Dus vergeet niet dat je het doet omdat jij onderzoek doen leuk vindt.



Interdisciplinary project SimBethel

Studying in the United States

Stijn Frima

It was in October 2011 that I was asked to participate in the SimBethel project. I have always wanted to study or work abroad and this was a good opportunity. For the SimBethel project we stayed in Davis, California, the United States for five weeks. I did this project together with three Civil Engineering students and one Architecture student. We were sponsored by the companies Deltares and HKV as well as the TU Delft organization DIMI.

First impressions

Davis was a real university town. It was not very large and therefore the university UC Davis had a big impact on the town itself. The UC Davis is a very green university and is comparable to the Dutch university of Wageningen. Therefore Davis was not a very American town. Many people in Davis have a bike, and cycle to their work or the university. They even have cycling lanes and the people I met were highly educated.

Although we worked in Davis our project was about the Bethel Island, a small Island in the Delta River, two hours away from Davis. Our goal was to create a board game about the dikes and general flood safety of the island. The game was a serious game. It had to teach the people of Bethel Island how strong their dikes should be and how they should maintain them. My job as a computer scientist was to create a computational model of the damage done to the island if a flood would occur. We Dutch are very used to water and what Americans call 'the concept of living under sea level'. They talk about it as if you would continuously be aware of the fact that you are now walking (or reading this article) under sea level. As if every moment the water could rise to astounding levels. Bethel Island was also under sea level and they had a very different approach to flood control. Their dikes for example were privately owned and the owners could do a lot with these dikes. These cultural differences made the project more interesting than if I would have gone on holiday to the United States, because you can only notice these things if you have studied them. Otherwise you would have gone to the tourist attractions and you would not have noticed these privately owned dikes (although, not all dikes are privately owned in the US).

Housing

For the duration of the project we were staying in a student apartment building. We had to arrange the housing ourselves and we were lucky that many student apartments were vacant. That also made it relatively easy to find a student apartment. We shared the shower and kitchen, but because there were so many vacant apartments that never lead to any problems. It was smaller than my room in the Netherlands and I had to pay more, but that is something you have to accept if you are going abroad.



Working in an interdisciplinary team

Of all the projects a student does at the Bachelor Computer Science, none of these projects are interdisciplinary or international. This project was both interdisciplinary and international. As a computer scientist I had to know a lot about civil engineering to even program the software. It was not only harder to communicate with the other students, it was also more difficult to program the software since there was knowledge which would have been very helpful to know.

It was smaller than my room in the Netherlands and I had to pay more,

but that is something you have to accept if you are going abroad.

Although interdisciplinary and international projects are important for the university on paper, the rules for a bachelor project make it very hard to do an international and interdisciplinary project in reality. My bachelor coordinator Martha Larson really made this possible. The problem is that although you are doing many things that are not computer science, your entire product has to be a computer science product which it is not, because it is an interdisciplinary project.

Students of other studies do not know what and what is not possible to accomplish with software solutions. Therefore I had to explain much more often that some of their wishes were not feasible. This is of course also the other way around, that I didn't know what was possible for other solutions. We first created a very complex serious game, half way through the project we deemed it too difficult and created a simpler version. Because the group is interdisciplinary we thought we had more expertise, but that did not necessarily mean we could create something more complex.



Whatever team members you have, it is very important to have an architecture or industrial design student in your team if you are creating a product and not only a report. The beauty of the product is of extreme importance and our serious game would have become very ugly was it not for our architecture student Eva Smits who did a great job.

Education

One of the advantages of our project was that we did not have to follow any courses or lectures. We worked on our project and talked mostly with researchers of the watershed sciences of UC Davis. Because we did not follow any courses we could skip many of the bureaucracy of applying to a different university. We had to fill in a few papers, but that was all.

Out of the comfort zone

Once I joined this project I had to do many new things that are out of your normal comfort zone. The realization that you even have a comfort zone is

The realization that you even have a comfort zone is one of the important lessons of such a journey.

one of the important lessons of such a journey. In my case I had to take the initiative to talk to companies and people, instead of those companies and people coming to me. We had to give many presentations and explain our product. We gave presentations for Deltares and HKV, for Arcadis in the United States, for students of UC Davis, at a Safety Fair in Bethel Island, at a meeting of the Municipality of Bethel Island and thanks to Deltares we could give a small demonstration of our game at the European FLOODrisk 2012 conference in Rotterdam.

We had meetings with Deltares and HKV, dinners with professors of UC Davis, dinners with a committee of the Municipality of Bethel Island and a levee patrol (where you inspect the dikes of the island) of Bethel Island, a boat trip around Bethel Island and a boat trip with a very generous professor of UC Davis.

We had two test sessions with professors and PHD students from UC Davis, many with our own team, with friends and with people from Bethel Island at a Safety Fair and with Deltares and HKV.

We had to work with extraordinarily strict deadlines and we had to advise people who were much older than we were and to people who did not care. I had to travel and work with a group I barely knew.

It makes you realize that many of your arguments not to do something are because it is new and different, not because it is a bad idea.

Traveling

After weeks of hard work, we finally got around to travelling in our last week. What I really wanted to do was visit UC Berkeley and Stanford. UC Berkeley and UC Davis have many similarities, they are both universities with a very 'green culture' and with many students advocating the importance of the environment. They both have many students from different cultures and are both public universities. Stanford however is a private research university and is very different. The university is huge and the student population is not as diverse as in UC Davis or UC Berkeley.

We also visited Yosemite National Park where we saw the giant Sequoia trees. After which we visited the desert Death Valley and Las Vegas. Las Vegas was not entirely what I expected. In American culture one also visits Las Vegas if you retire. And since we were not there in the holiday season we saw a lot of elderly people. Gambling is not the only thing you can do, you can see many beautiful buildings and in the evening there are many shows. For example, we went to the water show of the Bellagio hotel. If you go to Las Vegas: do not go there to gamble, but to see the beautiful hotels and observe all the crazy people.

Finally

Although I have spent more than the required bachelor project hours on the project, I still have not finished it. After I am finished with the bachelor project, we might continue with the project. We are thinking of producing the board game and to then attempt to sell it. Although it has been hard work, I am very grateful to have been asked to participate in this project.



Bethel Island, California



Historical Person: Edsger W. Dijkstra

Rebecca Jacobs & Frank Tabak

The historical person of this issue is our very own Edsger Wybe Dijkstra.

This Dutch computer scientist is mainly known for the development of the shortest path algorithm. His contribution to uprising of computer science in the Netherlands and the evolution of programming languages has been invaluable. We hope that after reading this article you can appreciate Dijkstra's views on programming and know exactly why he had a beard.

Dijkstra was born on May 11, 1930 in Rotterdam. He studied theoretical physics at Leiden University but soon realized that he had more affiliation with computer science. In 1952 he was employed as scientific programmer, the very first in the Netherlands, at the 'Mathematisch Centrum' which is nowadays known as 'Centrum Wiskunde Informatica' in Amsterdam. His task was to program the ARRA I (Automatische Relais Rekenmachine Amsterdam), the very first computer build in the Netherlands. In meanwhile he still was enlisted as student at Leiden University which he was until 1956.

The ARRA I

In 1956 with the introduction of ARMAT (Automatische Rekenmachine Mathematisch Centrum) Dijkstra decided to have an interesting demonstration, something that non-mathematicians can understand. He wanted that the machine was able to find the shortest path between two cities in the Netherlands in a simplified road map. For this he had to have an algorithm which solves this problem. He designed it when he was having a coffee-break from shopping with his fiancée. It took him 20 minutes and no pencil and paper to create this shortest-path algorithm which is widely used in our modern society. He waited three years, till 1959, before he actually published the algorithm in 'Numerische Mathematik'. That was only because the editor, F.L. Bauer, asked Dijkstra if he could contribute something. At the time an algorithm for the shortest path problem was hardly considered mathematics; there was a finite number of ways of going from A to B and obviously there is a shortest one, so what's all the fuss about?

In 1957 he married his fiancée Maria Debets with whom he eventually got three children: Femke, Rutger and Marcus. This marriage posed a problem when Edsger had to state his profession for the marriage record; 'programmer' was not accepted, 'theoretical physicist' was possible, however.

In 1960 Edsger Dijkstra was involved with the development of the programming language ALGOL-60, for which he worked on the compiler. During this work he had a deal with one of his colleagues, Jaap Zonneveld; they would but not shave till their work was done. Jaap shaved his beard afterwards, but Edsger fancied the beard and kept it for the rest of his life.

Dijkstra started programming on paper for machines that weren't built yet. As a consequence he had to be absolutely certain of the correctness of his programs, since he had no way of testing them. Dijkstra took writing very seriously; he had a very clear handwriting which would make sure that there could be no confusion as to what he programmed on paper. In the late nineties Luca Cardelli designed a font for Macintosh computers which was based on Dijkstra's handwriting.

Dijkstra viewed programming as composing music. Mozart allegedly composed his compositions first in mind and then would write down the composition in one go. Edsger programmed as Mozart, first he would think out an elegant solution to the problem in his head, and once he was fully confident about it he would write it down in one go. He emphasized the need of 'programming proofs'. In fact, Dijkstra never owned an own computer until the late nineties, then he got an Apple Macintosh from the university. He barely used the machine, and if he did, he would use it to send an e-mail.

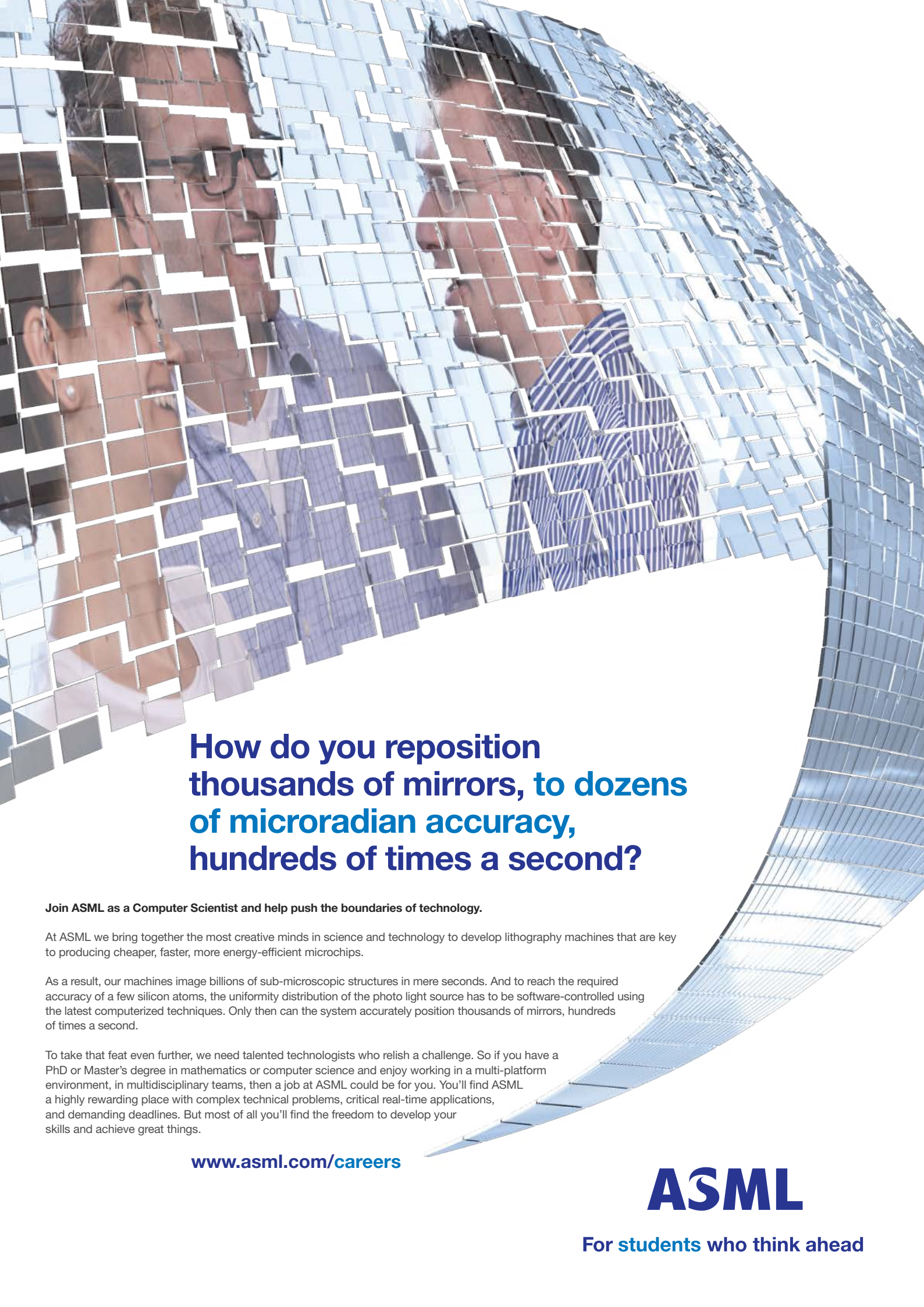
In 1962 Dijkstra got appointed as Professor of Mathematics at Eindhoven University of Technology. About this he once said: "Later I learned that I had been the Department's third choice, after two numerical analysts had turned the invitation down; the decision to invite me had not been an easy one, on the one hand because I had not really studied mathematics, and on the other hand because of my sandals, my beard and my 'arrogance' (whatever that may be)." He started building a team of computer scientist within the Mathematics department, since this and other universities did not have a computer science department yet. At this he developed the 'THE' (Technische Hogeschool Eindhoven) operating system for the Electrologica X8 computer. Many features of this operating system have become standard features in the operating systems we use nowadays. In 1972 Dijkstra was distinguished with the prestigious Turing Award for his contributions to computer science in the fifties, such as his contribution to the development of ALGOL-60. It was in 1984 that Dijkstra left Eindhoven and started working for the Computer Science department at the University of Texas. He already had close connections with this department for he had joined the Burroughs Corporation in Austin, Texas, as a research fellow in 1973. He remained at the University of Texas until he retired in 1999.

In the United States he and his wife bought a Volkswagen camper which they used to make long trips to national parks. This camper, which they called 'The Touring Machine', was used until 2001 when Dijkstra was diagnosed with bowel cancer. He and his moved back to the Netherlands. He died in August 6, 2002 in Nuenen. After his death the ACM renamed the PODC Influential Paper Award in distributed computing to the Dijkstra Prize in his honour. 

References

Alles moest nog worden uitgevonden: de geschiedenis van de computer in Nederland, Cordula Rooijendijk, Olympus, 2010, 978-90-467-4404-8

An interview with Edsger W. Dijkstra, Philip L. Frana, Thomas J. Misa, *Communications of the ACM*, vol 53, Issue 8, August 2010, pages 41-47, New York, NY, USA



How do you reposition thousands of mirrors, to dozens of microradian accuracy, hundreds of times a second?

Join ASML as a Computer Scientist and help push the boundaries of technology.

At ASML we bring together the most creative minds in science and technology to develop lithography machines that are key to producing cheaper, faster, more energy-efficient microchips.

As a result, our machines image billions of sub-microscopic structures in mere seconds. And to reach the required accuracy of a few silicon atoms, the uniformity distribution of the photo light source has to be software-controlled using the latest computerized techniques. Only then can the system accurately position thousands of mirrors, hundreds of times a second.

To take that feat even further, we need talented technologists who relish a challenge. So if you have a PhD or Master's degree in mathematics or computer science and enjoy working in a multi-platform environment, in multidisciplinary teams, then a job at ASML could be for you. You'll find ASML a highly rewarding place with complex technical problems, critical real-time applications, and demanding deadlines. But most of all you'll find the freedom to develop your skills and achieve great things.

www.asml.com/careers

ASML

For students who think ahead

SMART TRADING FAST TECHNOLOGY GLOBAL OPPORTUNITIES ARE YOU IN?

IMC FINANCIAL MARKETS STARTED ON THE TRADING FLOOR IN AMSTERDAM IN 1989, AS A MARKET MAKER IN DUTCH SINGLE STOCK OPTIONS. WHEN EUROPEAN EXCHANGES STARTED TO TRANSFORM INTO ELECTRONIC MARKET PLACES, IMC WAS AT THE CUTTING EDGE OF THIS TRANSITION AND ONE OF THE FIRST TO START ELECTRONIC TRADING.

TODAY WE ARE ONE OF THE WORLD'S LEADING PROPRIETARY TRADING FIRMS AND A KEY MARKET MAKER IN VARIOUS PRODUCTS

LISTED ON EXCHANGES THROUGHOUT THE WORLD. OUR EDGE COMES FROM KEEPING OUR TECHNICAL PEOPLE IN VERY CLOSE PROXIMITY TO OUR TRADERS: PUTTING THE BEST TECHNOLOGY AVAILABLE TO WORK AND COLLABORATING WITH EACH OTHER ON INNOVATIVE WAYS OF USING AND IMPROVING IT.

IMC EMPLOYS AROUND 600 PEOPLE AND HAS OFFICES IN AMSTERDAM, ZUG, CHICAGO, SYDNEY AND HONG KONG.

WE ARE LOOKING FOR:

► JUNIOR TRADERS

TO JOIN OUR EUROPEAN TRADING TEAMS.

► JUNIOR QUANTITATIVE RESEARCHERS

TO JOIN OUR EUROPEAN QUANT TEAMS.

► JUNIOR SOFTWARE ENGINEERS

TO JOIN OUR EUROPEAN ENGINEERING TEAMS.



OUR GLOBAL TRAINING PROGRAMS HAVE TWO FIXED STARTING DATES: FEBRUARY AND AUGUST. VISIT OUR WEBSITE FOR MORE INFORMATION AND TO EXPLORE THE CURRENT OPPORTUNITIES:
WWW.IMC.NL/FINANCIAL-MARKETS/CAREER-OPPORTUNITIES



imc

financial markets &
asset management