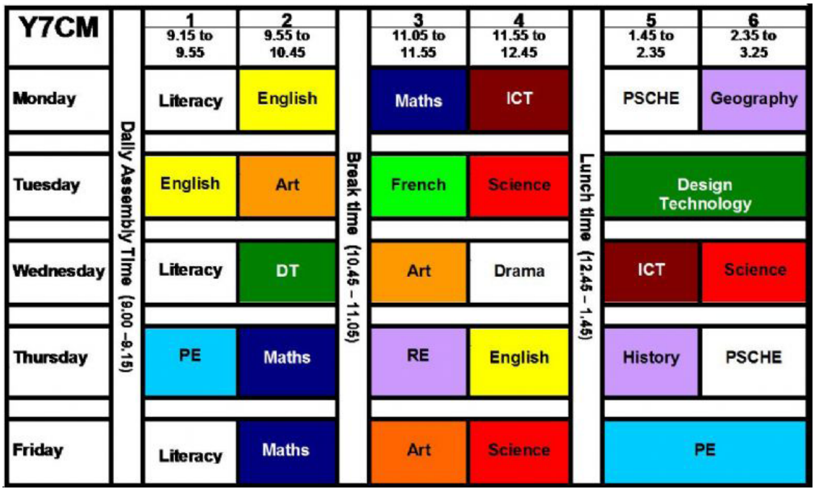
perio*diek

op regelmatige tijden terugkerend jaargang 2015 nummer 4

XX Hola a todos

In September last year Oscar went studying for half a year to Valancia. Find out about his experiences here and if studying abroad is something for you!



Y7CM		1 9.15 to 9.55	2 9.55 to 10.45		3 11.05 to 11.55	4 11.55 to 12.45		5 1.45 to 2.35	6 2.35 to 3.25
Monday	Daily Assembly Time (8.00 –9.15)	Literacy	English	Break time (10.45 – 11.05)	Maths	ICT	Lunch time (12.45 – 1.45)	PSCHE	Geography
Tuesday		English	Art		French	Science		Design Technology	
Wednesday		Literacy	DT		Art	Drama		ICT	Science
Thursday		PE	Maths		RE	English		History	PSCHE
Friday		Literacy	Maths		Art	Science		PE	

XX High School Timetabling

The optimalisation problem of finding the Perfect Timetable of a high school student. Go with Mirjam on the mathematical journey of finding this roster and discover both the theoretical and the pratical part of this problem.

30 Bitcoin

Is bitcoin the currency of the future, or is it simply a hype that will blow over? Discover how the bitcoin transaction system works and what the prospects of bitcoin are for the future.

In this Periodiek

From the editor in chief

Last year I went on excursion, the GBE, to Indonesia and Singapore. This trip was a really great experience and I made a lot of new friends. Now that I have been back for quite a while, having experienced the Dutch weather to the fullest extent, I still hold warm feelings to this trip. To welcome the international students going the other way, we changed the Periodiek to an international magazine. According to Steve Jobs: "Innovation distinguishes between a leader and a follower." With this change, the Periodiek will stay the leader in delivering news on the research of this faculty and everything you always wanted to know.

When we changed the language to English, we also changed the looks of the Periodiek a bit. You might notice the use of more colour. We hope that you like the new look as much as we do. I feel this change gives new energy to continue our ambitions, to lift the Periodiek to the next level. I hope that these Christmas holidays will also give you the time to innovate your personal life. To become a leader of your own life, to gain new energy to fulfil all of your ambitions.

— *Douwe Visser*

Editors Derk Rouwhorst, Douwe Visser, Klaas Hakvoort, Nick Lutjes, Thomas Meijknecht.

Authors Inne Lemstra, Mirjam de Vos, Jos Borger, Oscar Delicaat, Martine Schroor.

Advertisers ASML (p. 9), Talent & Pro (p. 15), Topicus (p. 20 - p. 21), KxA (p.29).

Advertise? Contact us at bestuur@fmf.nl.

Print run 1200 pieces

Press Gildeprint

ISSN 1875-4546

The Periodiek is a magazine from Fysisch-Mathematische Faculteitsvereniging and appears five times per year. Previously printed Periodiek's can be found at perio.fmf.nl. The board of editors can be reached by perio@fmf.nl.

In the news

First successful rocket landing on earth

The spaceship company Blue Origin had one of their carrier rockets make a controlled landing. It is the first time that a rocket lands right-side-up.

Blue Origin launched the New Shepard-capsule on a BE-3 rocket. This combination reached a hight of 100 kilometres. At this hight, the capsule was detached, allowing both capsule and rocket to make a decent. The BE-3 carrier rocket activated its booster engines at 1.5 kilometres up, reducing the speed of the rocket on impact to just 7 kilometres per hour.

Companies like Blue Origin and SpaceX want to make space exploration cheaper and more sustainable. Allowing rockets to have a controlled landing back on earth, and thus make the rockets reusable, is an important step in achieving this goal.

scientias

Scientists breed mosquito to combat malaria

Scientists can breed a species of mosquito that can not spread malaria. Biologists hope to combat the spread of malaria by introducing these mosquitoes into nature and allowing them to mix with species that can spread the disease.
When announcing the discovery, the scientists said that they had focused on the anopheles stephensi, a species that is responsible for spreading malaria in India. By altering the DNA of this species, 'malarialess' mosquitoes now mix with their dangerous congeners. Biologists aim for 99.5% of the descendants of the new species to be not able to spread malaria to humans.
Biologists don't think that this new method can exterminate all malariamosquitoes, but could be very good against the disease. According to the World Health Organisation, this year 214 milion people will contract the disease and 438,000 of them will die.

nu

Exploring the physics of a chocolate fountain

A mathematics student has worked out the secrets of how chocolate behaves in a chocolate fountain, answering the age-old question of why the falling 'curtain' of chocolate surprisingly pulls inwards rather than going straight downwards.

"Chocolate fountains are just cool, aren't they!" says Adam Townsend, an author on the paper. "But it's also nice that they're models of some very important aspects of fluid dynamics." The conundrum of the converging curtain was solved by looking at some classic work on 'water bells'.

The physics of the water bell is exactly the same as the falling curtain of chocolate, and the reason the chocolate falls inwards turns out to be primarily surface tension.

They also looked at the flow up the pipe to the top of the fountain, and the flow over the plastic tiers that form the distinctive chocolate fountain shape.

"Both the chocolate fountain and water bell experiments are surprisingly simple to perform," Dr Wilson continues, "however they allow us to demonstrate several aspects of fluid dynamics."

"It's serious maths applied to a fun problem." continues Adam Townsend. "I've been talking about it at mathematics enrichment events around London for the last few years. If I can convince just one person that maths is more than Pythagoras' Theorem, I'll have succeeded. Of course, the same mathematics has a wide use in many other important industries, but none of them are quite as tasty as chocolate."

phys.org

First successful rocket landing on earth

Army ants are nomads and regularly track through the rainforest. Such a track does not go easily, the ground in a rainforest is very uneven and once in a while the ants hit holes in their route. Instead of walking around such an obstacle, the ants built a bridge out of their own bodies, allowing other ants to walk across them. The bridges appear and disappear within seconds and allow the ants to quickly move across unknown and unpredictable territory.

Researches have now discovered that these bridges move. Previously, it was thought that once build, these bridges were static structures. However, the bridges slowly move away from their starting point to create a shorter route. The bridge also widens if there is a lot of traffic. It is all very special, says researcher Cristopher Reid. "Imagine that you could relocate the George Washington-bridge between New York and New Jersey to adapt to the direction of rush hour traffic."

The bridges are the result of a strong piece of collaboration. The researchers want to use these findings in ants in order to convince robots to collaborate equally well. Examples are exploration or rescue missions. "Such swarms of robots can do remarkable things, like creating bridges to cross complex terrain," says Reid.

scientias

AUTHOR: **INNE LEMSTRA**

Introducing the new board

A new year, a new board. An association is built on everyone that participates. A little steering and coordinating of all these people is usually quite welcome. Filled with people eager to determine the fate of the FMF, this is the purpose of the board. But who are these individuals, with virtually unlimited power, that control everything from the shadows? As the secretary of the FMF I have the honour of introducing my fellow board members. If I had to describe our board as a whole, I would say our average age is quite high, so we hope to create great policies using our combined life's experience.

Jos Borger - Chairman

The smallest of the current board members with his 1 metre and 68 centimetres. Do not be fooled though, Jos is not a pushover. With mixed martial arts as his hobby, it takes quite a considerable force to keep Jos from executing his chairman duties. He is usually the social type, however Jos can be surprisingly nerdy when it comes to initiatives like videogames or LED-cubes. Being in the organising committee of the GBE, the great foreign excursion, to Indonesia and Singapore, this small man has experience handling big jobs. To find out what his experiences of being a FMF board member are so far, you can read his column, which is also in this Perio*diek.

Emily Mook - Vice Chairman & Commissioner of Education

This ambitious power woman knows exactly who she is and what she wants. She is really passionate about improving the education standards for students. With this she fits the recently formed role of commissioner of education well. A goal she is trying hard to achieve is setting up catch-up sessions, wake-up calls for students that have fallen behind on a course. Another thing she wants to achieve is becoming a person that can guide students with questions to the right people to talk to. I am glad that Emily is on our team, since this fierce lady gets what she wants, when she wants it. Fortunately Emily also has a feminine side, she enjoys cooking a fair bit, with a preference for baking cakes.

Thiadrik Tiesma - Treasurer

The person that I have known the longest of my fellow board members. With different committees under his belt, FMF'ers do not come more hardcore than this. Certified beard owner and professional Frisian. On Thiadrik you can count when it comes to defending his morals as well as the FMF budget. Thiadrik has quite some experience regarding the general musings of the association. Sometimes he can be a bit bearish, but you know he is not really going to hit you with that axe, right, right? A hidden side of Thiadrik is that he owns a slow cooker, with which he likes to prepare the most amazingly delicious meat dishes. Thiadrik is known to enjoy his metal music as well as his videogames.

> *"Who are these individuals, with virtually unlimited power, that control everything from the shadows?"*

Jim Baarslag - Commissioner of External relations

New to the study association, Jim quickly climbed the ladder of FMF stardom. Being kind and respectful to those that require his assistance, often times providing practical solutions to the problems at hand. Jim can come off as quite easy-going, however underneath his laid back demeanour lies someone who is quite the professional. He has a hidden impulsive side to him, if Jim wants things done he gets things done. This has lead, for instance, to a drone suddenly flying through the NSFW. He and Jos have been friends since the start of their studies. He gives a new perspective on things, I for instance was able to add a few words to my vocabulary thanks to Jim.

Nick Lutjes - Commissioner of Internal relations

Nick is actually the perfect stranger of the FMF. This in the sense that he only actively joined in the fourth year of his studies, but he fits the FMF well. Nick is diligent and quite a hard worker, this he needs to lead the ever growing number of committees within the FMF. Thankfully he has gained some valuable diplomatic experience doing the LANcie. He was even chairman of the MegaLAN, the joint LANparty of four different study associations. This give him some valuable managing and negotiation skills. In his free time Nick likes to game, but he also likes to dive.

Inne Lemstra - Secretary

As the oldest of the board, I bring a lot of experience to the table. Even though I am studying molecular biology, a study that is not part of the studies the FMF represents, I still feel like a real FMF'er. I joined the FMF very early on in my studies, at that time I still studies applied physics, and have participated and organised many activities. I even won the title FMF'er of the year, via the May-Month-FMF-Month-challenge, twice. The Bakborrel is another event I really cherish, it sparked my love for baking and every year I am a fierce competitor. I see myself as a motivator and someone that can provide original ideas, some good, some very bad. Outside of the FMF, my hobbies are playing videogames, with people of the FMF, and Kendo, which is Japanese sword fighting •

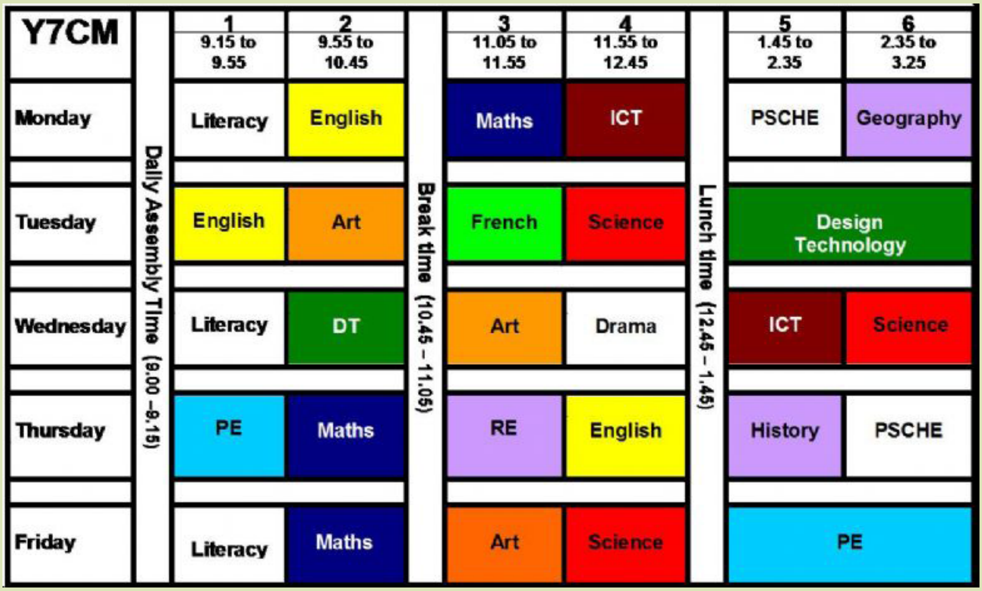
RCEPTIONS
AL OF INTERNATIONAL AFFAIRS
Power Shifts in East Asia: Balance of Power vs. Liberal Institutionalism
Namrata GOSWAMI
EU- East Asia Relations: Effects of the Global Financial and Eurozone Debt Crises
Ramon PACHECO PARDO
India and East Asia: The Look East Policy
G. V. C. NAIDU
Brazil, East Asia, and the Shaping of World Politics
Alexandre UEHARA & Guilherme CASARÕES
Chinese Foreign Policy as a Rising Power to find its Rightful Place
Suisheng ZHAO
From Engagement to Contention: China in the Global Political Economy
Sadık ÜNAY
Making Sense of 1 March: A Proactive Strategy of Avoidance
Hasan B. YALÇIN
The Turkey-Armenia-Azerbaijan Triangle: The Unexpected Outcomes of the Zurich Protocols
Zaur SHIRIYEV & Celia DAVIES
Turkish-Armenian Normalisation and the Karabakh Conflict
Cory WELT
Book Reviews
perio*diek
PCM
COMPLETE INSTALLATIEGIDS VO
WINDOWS 1
SUPERSNEL UPGRADEN
DE BESTE HARDWARE
10 SUPERTIPS
APK VOOR JE NAS
DRAADLOOS STREAMEN
SMART TV'S
TECHNIEK IN WEARABLES
ACER REVO ONE
PRINT QUARTERLY

AUTHOR: **JOS BORGER**

Perio*diek in the university library

Not so long ago I was in the university library to print something, when I saw the collection of journals on the first floor, in the place also known as the aquarium. Cabinets filled with scientific journals and at times an association magzine. When I saw this, I imediatly walked to the letter P, searching for our precious "Perio*diek", but found that it was not there. Knowing the Perio*diek to be of high quality, especially compared to some of the association magazines I saw standing there, I assumed it was possible for our magazine to be included in this collection and so I tried. A few emails and another visit to the university library later this was a fact. I am proud to say that the Perio*diek is now part of the magazine collection of the university library of the University of Groningen. All issues of the academic year 2014-2015 and all issues to come can now be found in the university library. So if you are busy studying in the library and are in desperate need of some relaxing yet stimulating distraction in the form of a high quality association magazine, visit the collection on the first floor and be amazed by this wonderful magazine •

*"I am proud to say that the Perio*diek is now part of the magazine collection of the university library of the University of Groningen."*



Y7CM		1 9.15 to 9.55	2 9.55 to 10.45		3 11.05 to 11.55	4 11.55 to 12.45		5 1.45 to 2.35	6 2.35 to 3.25
Monday	Daily Assembly Time (9.00 –9.15)	Literacy	English	Break time (10.45 – 11.05)	Maths	ICT	Lunch time (12.45 – 1.45)	PSCHE	Geography
Tuesday		English	Art		French	Science		Design Technology	
Wednesday		Literacy	DT		Art	Drama		ICT	Science
Thursday		PE	Maths		RE	English		History	PSCHE
Friday		Literacy	Maths		Art	Science		PE	

AUTHOR: MIRJAM DE VOS

High School Timetabling

In addition to algebra, analysis and many other disciplines, optimization is also a part of mathematics. This discipline has overlap with econometrics and operations research. A subdiscipline of optimization is timetabling. Timetabling consists of scheduling sporting events, work shifts, and education timetables. Every part has its own distinct properties, so a variety of solution methods is needed. In this article we will have a look at the problem of high school timetabling, both in theory and in practice.

A Hard Optimization Problem

Every country has its own rules concerning high school education. While junior and senior grades in Greece are generally scheduled separately, in the Netherlands teachers often teach to both lower level and upper level classes. In one country a student can only choose between ‘difficult’ and ‘less difficult’ mathematics and between language A or B, while in another country students have to choose a track, courses within this track, and additional courses.

Difficulties in Dutch High School Timetabling

In Dutch high schools there are two factors that make the timetabling problem extra complicated. The first factor is that many different combinations of courses can be chosen in the upper level grades. A student chooses one out of four tracks and a number of electives. Sometimes the list of available elective courses just contains all courses offered at the school. In this way, every student could choose a different curriculum. To allow every student to follow every course in his/her curriculum, many time intervals are necessary.

As a consequence, their timetables will contain many idle times.

The second reason why timetabling on Dutch high schools is so difficult, is because of the large number of part-time teachers. According to the collective labour agreement, a part-time teacher only has to work a fixed number of days per week. If this applies to one teacher there is no problem, but when more than half of the teachers have such conditions associated with them, it becomes quite difficult. On top of that, teachers are often allowed to register their preference for days off. Suppose all teachers have one or two preferred days off, most of them choosing Monday and Friday. Then try making a timetable that looks good for the students …

NP-Complete

Before we take a look at the way we can formulate the high school timetabling problem mathematically, we should note that the problem is NP-complete [1]. It is difficult to explain what this means in a few sentences, but I will try.

Some problems are 'easy'. These can be solved by a polynomial algorithm, i.e., an algorithm of which the running time can be expressed as a function that is polynomial in the amount of variables. These problems are contained in complexity class P. For other problems, there is not yet a polynomial algorithm found which solves it optimally. However, we might be able to check in polynomial time whether a potential solution really is a solution. If an algorithm which does this check exists, the problem is in NP. One of the so called Millennium Prize Problems is the problem whether P equals NP. If P equals NP, then there exist polynomial algorithms to solve all problems in NP.

A problem which is NP-complete, needs to be in NP. The timetabling problem we will describe is NP-complete. Within a limited timeframe we cannot find a solution proven to be optimal, but we can go looking for good solutions. An algorithm which can do this is called a *heuristic algorithm*, like the nearest neighbour algorithm for the travelling salesman problem.

High School Timetabling Problem as Linear Program

We need variables that indicate at what time which class will be in the timetable. The set $\mathcal{E}$ of *events* is used, which consists of all classes to be scheduled. There are for example three geography classes in this set if this course is offered three hours per week. There

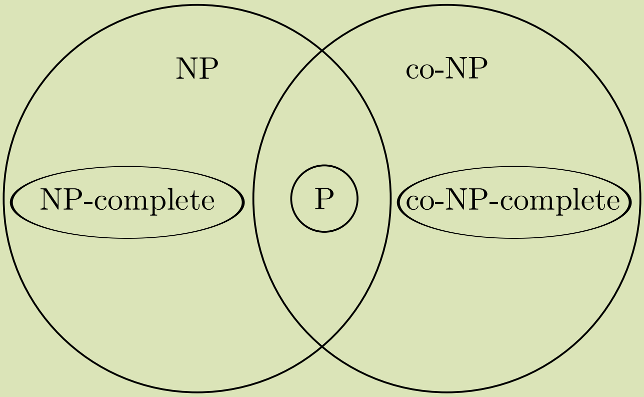


FIGURE 2 The relation between the different complexity classes.

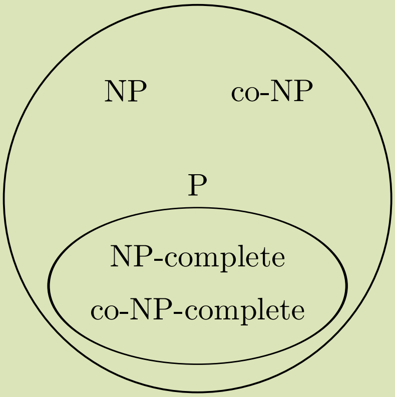


FIGURE 3 The relation between the different complexity classes if P= NP

Linear Programming

Every timetabling problem can be formulated as a linear program. A linear program can be solved by programs such as CPLEX or Gurobi. We need variables which describe what problem needs to be solved (decision variables). In addition there is an 'objective function' which describes what will be maximized or minimized. Lastly, there is a number of constraints which indicate requirements and wishes regarding the problem. A linear program can look like this:

$$\max c^T x$$
$$\text{s.t. } A^T x \leq b$$

The matrix $A \in \mathbb{R}^{m \times n}$ and vectors $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$ are given, while $x \in \mathbb{R}^n$ is the vector consisting of variables that describe the solution corresponding to the instance. The part we maximize, $c^T x$, is called the *objective function* and c_i is called the *cost* of variable x_i. A single constraint has the form $a_i^T x < b_i$, where a_i is the i^{th} column of matrix A. The *feasible set* or *solution set* $\mathcal{F}$ for a linear program is the set of all x which satisfy all constraints:

$$\mathcal{F} = \{x \in \mathbb{R}^n \mid A^T x \leq b\}$$

Any $\overline{x} \in \mathcal{F}$ is called a *feasible solution* and $c^T \overline{x}$ is the *objective value*. $\overline{x}$ is an *optimal solution* of the LP if it maximizes the objective value, i.e. if there is no $x \in \mathcal{F}$ such that $c^T x > c^T \overline{x}$.

also exists a set $\mathcal{T}$, consisting of all times in the week at which a class can be scheduled. The most important (binary) decision variable is defined as follows for events $\forall e \in \mathcal{E}$ and times $t \in \mathcal{T}$:

$$x_{e,t} = \begin{cases} 1 & \text{if event } e \text{ is planned to start at time } t \\ 0 & \text{otherwise} \end{cases}$$

There is a number of requirements connected to this variable, for example that a single event can never be scheduled at two different times. This we formulate as:

$$\sum_{t \in \mathcal{T}} x_{e,t} \leq 1 \quad \forall e \in \mathcal{E}$$

The sum of all decision variables connected to a certain event has a maximum of 1. As $x_{e,t}$ is a binary variable, it can only be 0 or 1, so $x_{et}=x_{e't'} = 0.5$ does not occur in any feasible solution.

In addition to the variables, we also have to define an objective function and extra constraints. The constraints describe the requirements and wishes imposed on the timetable. A requirement is a *hard constraint* and a wish is a *soft constraint*.

Objective

The goal is of course to obtain a timetable with all classes scheduled and meeting the requirements. Our objective will maximize the number of scheduled classes. This means a timetable in which not all classes are included is also feasible. If this would not be the case, there would be no solution for a program that results in a timetable where one class is not scheduled.

If there are soft constraints we have to subtract something from the objective. A timetable that does not meet a wish defined via a soft constraint is feasible, but gives a lower objective value than a timetable that does meet the wish. This gives us for example the following objective function:

$$\max \sum_{e \in \mathcal{E}} \sum_{t \in \mathcal{T}} x_{e,t} - \sum_{c \in C_{AUT}} \sum_{r \in c} w_c \cdot s_{c,r}$$

What the second part of this formula means will be explained later.

Avoid Clashes Constraint

There are all kinds of constraints that set requirements or wishes concerning the timetabling problem. An example is the requirement that two classes given by the same teacher or followed by the same group of students cannot be scheduled at the same time. In order to implement this requirement it is useful to add an extra variable. This integer variable $y_{r,t}$ indicates the number of events that *resource* r has on time interval $[t, t+1)$. The collection of all *resources* contains all students and all teachers. This is defined as follows, with E_r being the collection of all events attended by a resource r and I the collection $I = \{i \in \mathbb{N} \mid 1 \leq i \leq D_e,\ t-(i-1) \geq 0\}$, with D_e

the length of *event e*:

$$y_{r,t} = \sum_{e \in E_r} \sum_{i \in I} x_{e,t-(i-1)} \quad \forall r \in \mathcal{R}, t \in \mathcal{T}$$

The definition is added as constraint. Together with the constraint $y_{r,t} < 1$, this guarantees that a feasible solution will not contain any clashes.

Avoid Unavailable Times Constraint

As mentioned before, the large number of part-time teachers is one of the difficulties in timetabling for Dutch high schools. Suppose a teacher has young children, making her unable to teach first period and causing her to prefer to have the Wednesday afternoon off. The school board can set this as a requirement (hard constraint), which the timetable has to meet. However, it can also be a wish (soft constraint). In that case, a timetable in which the teacher's wish is not realized is also accepted. For every teacher r that is not available at a certain time, we formulate this constraint as follows:

$$\sum_{t \in c} y_{r,t} \leq s_{c,r}$$

We look at all times t that are contained in the constraint c, so the times at which the teacher is not available. The number of these times at which a class by teacher r is planned has to be smaller than a certain number $s_{c,r}$. This number is also part of the objective. This so-called *slack variable* $s_{c,r}$ will be minimized, because $-w_c \cdot s_{c,r}$ will be maximized. Every soft constraint c has a certain weight w_c. If we think the requested day off of teacher r_1 is more important than that of teacher r_2, then we make a new constraint c_1, for which w_{c1} is bigger than w_{c2}.

If the school board decides it is a requirement that teacher r is not scheduled at times he/she is not available, the slack variable $s_{c,r}$ is replaced by 0. Now for this teacher the right periods will remain free in every feasible solution.

In addition to the constraints described above, we can of course add many more wishes and requirements. Dr. Jeffrey H. Kingston describes a large number of constraints on his website [2]. Most of these constraints can also be found in my thesis [3]. In several countries research groups are busy developing new algorithms that make better timetables in less time. In order to compare the algorithms there exist benchmark instances [4].

In Practice

Since my graduation I work at the timetabling software company *Zermelo Roostermakers* and offer support to timetablers who work at high schools in the Netherlands.

The software deals with requirements like the Avoid Unavailable Times constraint slightly different than the way we described above. Per teacher we indicate the times she wants to have off using a penalty system. The more important achieving this is, the higher the penalty. If someone really needs to be free at a certain time, we put the penalty on that time at 1,000,000. This does not mean there will never be a class at that time though. The only way to enter this as a requirement is to fix a (dummy) class at that particular time. However, the penalty of 1,000,000 is preferred to a fixed dummy class. It makes it easier to find a feasible solution and start optimizing that solution.

To obtain a good timetable, the timetabler often starts with *clustering*. In a nutshell, this means that groups are made for which the classes can be at the same time in the timetable. This creates a sort of pre-timetable, which can in principle go into the timetable without using too many positions. Subsequently all kinds of algorithms are used to actually schedule the classes. This can be done in a variety of ways, but no single one is perfect. An optimal timetabling strategy can only be found if $P=NP$ •

References

[1] C. H. Papadimitriou and K. Steiglitz, Combinatorial optimization: Algorithms and complexity, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1982.
[2] http://sydney.edu.au/engineering/it/~jeff/hseval.cgi?op=spec&part=constraints
[3] M.A. de Vos, Solving the Dutch High School Timetabling Problem using Linear Optimization, http://irs.ub.rug.nl/dbi/5375d52c603be
[4] http://www.utwente.nl/ctit/hstt/

AUTHOR: **JOS BORGER**

From the President

My name is Johannes Christian Borger, but everyone calls me Jos. This year is already my sixth year studying in Groningen. I am the proud owner of a Bachelor of Science in Theoretical Physics and a Bachelor of Arts in Philosophy. I am currently in the second year of my masters Theoretical Physics: Quantum Universe and if everything goes according to plan, I will be studying for another year and a half. Studying for seven years might seem like a long time, but at least the first five years flew by. I am still not sure if I want to work in science or business.

I was born in the Martini hospital in Groningen on 11 March 1992. Before I started studying, I moved 12 times, some of the places I have lived are Atlanta (Georgia), Zuidhorn, Oestgeest, Utrecht, Ndounge (Cameroon), Niekerk, Almere and Amersfoort, where I have lived during my whole high school. I have one sister who also studies in Groningen. This year she is president of the Mattekloppers, the judo, karate, etc. association for students in Groningen. My sister and I both have judoed since 2001; I even did a national championship (NK) in my prime.

The first time I came into contact with the FMF was of course during the Pienterkamp, a fun weekend that yielded a lot of good memories and some of my first study contacts, some of which I still see. My second encounter with the FMF was some time later, in my second year, when the Huygens committee organized an excursion to CERN. As an aspiring physicist, I was excited by this trip and luckily, I got to go. This was a fascinating trip, not only did we visit some of the most cutting edge experiments, we also got a taste of the hard-core research atmosphere and even had time to enjoy ourselves during the free evenings. Overall, I was very impressed by the organization of this trip, especially because it was completely organized by students. Years later, Steven van der Veeke, one of my Pienter mates, asked me if I wanted to help organize the GBE, the great foreign excursion. I joined and became business commissioner. This was my first experience with doing committees and I really enjoyed it. Doing a committee makes you do all sorts of different things that you would not do during your regular study. The excursion went to Singapore and Indonesia and was a great success.

After this, I was not done with organizational work and not done with studying; this made me apply for the function of President of the FMF.

At the moment of writing, we are officially the board for two months, time goes remarkably fast if you are busy. I can only say that I think things are going very well. We start getting the hang of things and although there are always some things that go wrong, overall, we are doing very well, but that is only my humble opinion. I think this was a sufficient introduction of who I am, if you have any questions please ask them.

I hope to see you all during this academic year, in our room and during our activities. Have a good year •

talent&pro

Gezocht: bèta's in het bedrijfsleven

Via Talent&Pro krijg je de kans het beste uit jezelf te halen.

Wil jij jouw bèta-talent toepassen op vraagstukken in het bedrijfsleven? Dat is precies wat je in het actuariële traject bij Talent&Pro. Complexe berekeningen en analytisch vermogen zijn nodig bij vraagstukken als de woekerpolissen en het nieuwe pensioenakkoord.

Of wil jij liever werken op het snijvlak van bedrijfskunde en IT? Kun jij bruggen slaan tussen de gebruikers en programmeurs van informatiesystemen? In het business IT traject van Talent&Pro ga je onder andere aan de slag met grote data-analyses, procesoptimalisatie en automatisering.

Banken, verzekeraars en pensioenfondsen kunnen jouw hulp goed gebruiken. Via Talent&Pro doe je verschillende uitdagende opdrachten bij deze financiële instellingen. Bij Talent&Pro staat je persoonlijke ontwikkeling centraal: we bieden coaching en opleidingen zodat jij het beste uit jezelf kunt halen!

talent-pro.com

AUTHOR: **OSCAR DELICAAT**

Hola a todos!

April of last year I was struggling to obtain my driver's license. Many hours in with my Turkish teacher, who I would not recommend anyone, I was talking with a girl, who told me she was going to Lissabon to study. I always thought that it was hard to get into a foreign university and had never really considered going abroad before. The following year however, I would be in my third year of theoretical physics, which meant I was free to choose whatever I wanted.

After a talk with the study-advisor, all my worries about how hard it would be to get in were gone. The logistics boiled down to finding the very best city, filling in a double-sided contract to obtain a grant for €200,- a month, and picking a few subjects for evaluation by the exam-committee. My conditions for a city were not too strict: a warm country, with a beach at walking distance, and where the people speak Spanish. Valencia was the only university with which the Rijksuniversiteit Groningen had any relations that matched these criteria. I talked to some people who did their Erasmus there and was quickly convinced that this would be my destination.

I booked a cheap Ryan-air ticket and at the beginning of September, I set foot in Valencia. My preparation was limited to having joined a few Facebook groups and talking a little Spanish. I had already booked a hostel but I soon found out that I was too overconfident in my Spanish and that the average Spanish senior did not understand a word of English. That is why it became an expensive ride to my first hostel.

The next day I woke up and set some meetings to look at rooms that were offered in the Facebook groups. I checked out three shabby houses and decided that I would just need to meet someone who had a free room. This worked out really well, that night I went to an Erasmus dinner and was offered multiple rooms, all worries gone! I moved into a nice flat on a ten-minute walk from the beach, which is all I could wish for. I had one German roommate, who was doing his internship there, one guy from the south of France, who went to the same university as I did, and a Spanish guy in his late twenties, who worked all day and on the weekends went to his girlfriend, so he wasn't around much.

The start of the year was a great period. There were around five thousand foreign students and the beautiful thing is that everyone arrives on their own and all are eager to meet new people. You would drink one beer at night and go to the beach together the next day. It was two weeks until the lectures would start and it was still a lovely 35 degrees. The days were spent on the beach playing beach volley and ordering some beers from the beer runners there. At night we would discover the delicious tapas and paella's, the paella being thé dish from Valencia and you won't find it better elsewhere. Then at about 3 am the clubs would finally open their doors and the night could start. It is a completely different nightlife then here. Valencia has a great nightlife; it is also known throughout Spain for its humongous clubs. In the weekends, I frequently travelled around. Sometimes I went with the trips offered by Erasmus, other times I used Blablacar, paid car sharing basically, which was a good way to practice your Spanish.

> *"The days were spent on the beach playing beach volley and ordering some beers from the beer runners there."*

When the lectures started, I found out that I was not enrolled in a single class that I signed up for back in the Netherlands. It did not however surprise me, since I had gotten to know their great bureaucracy. The university was not too strict though, they told me to go to the classes, and that we would figure something out. Since I had decided that I wanted to turn my minor into a relaxing period, I had chosen to do half a year of economics. The university is not listed all that high in the rankings and I took my courses in English, so it was all fairly easy. The class was a nice mix of Erasmus students and Spanish students and the professor was up on a stage. The fact that the teacher was up on stage, I found quit remarkable, but also resembling the Spanish culture. In Holland, we all sit upstairs, look down at the teacher who tells his story, and while sitting in the lecture halls no one will tell you what to do except to shut up. In Valencia, the teacher spent the first lecture explaining their do's and don'ts and demanded absolute obedience. It was still common to be kicked out of class for talking or not paying attention. It may sound like a bad time, but if you just behaved the teacher were cool, lenient and taught you some interesting stuff. Besides, if you did not enjoy it, the cafeteria started selling beer from 12, so you always had an alternative. There was however also another university in Valencia for the natural sciences. This university was recently built and is a very modern university, so for all the students whom I scared off, there is a good alternative for studying as well!

Valencia soon felt like home, the week days were spent on the beach, with an occasional visit to the university and many nights drinking beers in Natura Pub, the best pub in town, where the flats around were full of banners with text such as "quieremos dormir" (we want to sleep). The Sunday's were spent windsurfing on the Mediterranean Sea. I went there together with my French roommate every weekend, where a weed smoking hippie in cowboy boots taught us windsurfing. He did not touch the water in all those lessons, so progress was quit mediocre but enjoyable nonetheless. The rest of Sundays were spent with friends enjoying paella or 'pollos asados' from El Rostidor, which we got from across the street. It was a family store, which only opened on Sundays, selling the best food in town.

Closing in on the end of the year, the weather started to get worse, meaning it dropped to around 20 degrees and the people started wearing scarves and winter coats. Around this time, a few friends managed to get their hands on some cheap tickets and flew over

to visit me over the span of a few days. It was great to see them again, although it took a while to get used to speaking Dutch again, and it inspired them to look into going abroad as well. One of them recently moved to Mexico, so I guess that was a success. I flew back to the Netherlands to spent Christmas and New Year's Eve with the family.

A few days after I got back, the exams were due, which meant that all excitement was on hold for a couple of weeks. Days were spent in the library and my German roommate had already returned home, so the house felt empty. Still, the weather was returning and I met a cool Polish girl, so the days were not all bad. The exams were not that hard and I even ended up being given the title 'estudiante notable', which was against all odds.

After these quiet weeks, I decided, with a couple of friends, that we needed some excitement. We signed up for a twelve day Erasmus trip to Morocco for a bargain price. We were one big group with all kinds of nationalities and went by bus from Spain via Gibraltar to Tangier in Morocco. From there, we drove through all of Morocco from Chefchaouen, over the Atlas Mountains to the Sahara, where we spent an unforgettable night with the Berber folks in the middle of the desert. Subsequently we continued our trip via the coast of Morocco via Marrakesh back to Tangiers (where we found eight refugees in a cylinder block) and 12 days later, we returned happy as can be, back to Valencia. It was an amazing trip and I made some great friends. It is this kind of adventures that you have, only when studying abroad.

Back in Valencia, my excitement was short-lived, because the realization kicked in that I was to leave Valencia five days later. I had some great dinners and good parties to finish my Erasmus. Almost all my friends were to stay for a complete year and I still find it a pity that I could not stay for the whole year.

> *“I hope that I inspired some of you to take on the adventure abroad and experience a great experience, saludos!”*

I got back to the Netherlands, were I had to live with my parents for the first month and a half. This was quite a big change of pace. Fortunately, I rapidly found a new room to move into. It was also nice, however, to hang out with my friends and actually study some relevant stuff again. I flew back over to Valencia to visit my friends again and enjoy the beach one last time but that was it.

It was the best experience during my university time and I would recommend it to anyone. I learned Spanish, probably met people from every country in Europe and from lots of countries outside Europe. Everyone has cool and interesting stories to tell and you learn that there are some completely different cultures out there. The best thing I learned from studying abroad was to be spontaneous, break away from monotone daily routines and to have ‘yes’ ready as your standard reply. I agree, it sounds lame and cliché, but they are some nice habits to have •

Effectieve visualisatie van bestuursinformatie

AUTEUR: GUIDO VAN CAPELLEVEEN, INFORMATIEANALIST (TOPICUS ONDERWIJS)

Waarom moet ik mijn gegevens registreren, mijn dokter weet toch wel wie ik ben? Dit zou je opa of oma zomaar gezegd kunnen hebben toen je daar op bezoek was. Vertrouwen en persoonlijk contact stonden vroeger hoog in het vaandel, tegenwoordig leggen we alles liever in systemen vast en delen we al deze informatie. We doen dit onbewust, elke dag, maar waarom eigenlijk? EPD, Leerlingvolgsysteem, hypotheekaanvragen, zelfs de vlokkentest ligt ergens in een systeem vast.

De één zal argumenteren dat het sneller werken is met de computer, de ander ziet het voordeel van het eenvoudig kunnen delen of versturen van informatie. Zelf kan je waarschijnlijk ook tal van redenen verzinnen waarom het voor jou van belang kan zijn om bepaalde informatie vast te leggen. Wie bepaalt eigenlijk dat dit goed is en welke afwegingen worden gemaakt bij het opstellen van het ontwerp en invoering voor dit soort systemen? Ervaren de gebruikers het systeem als een last of juist als een uitkomst? Met veel belanghebbenden bij informatiesystemen zie je vaak dat er geen ideaal is dat aansluit op ieders wensen, maar dat er wordt gezocht naar een situatie waarin alleen de informatie die van belang is voor het werkproces aanwezig is. Het is daarbij zoeken naar het kunnen dekken van de gemeenschappelijke informatiebehoefte, waarbij werkprocessen marktgebonden getracht worden te gegeneraliseerd. Processen beperken zich ook niet enkel tot één organisatie, maar vragen vaak een breder perspectief. Deze vorm van informatisering wordt ook wel keteninformatisering genoemd. Al deze gegevens creëren vervolgens een bron van nieuwe informatie die gebruikt kan worden voor de procesoptimalisatie of voor de processturing. De eerste vorm levert nieuwe inzichten vanuit de data op waarmee processen kunnen worden doorontwikkeld. De tweede vorm van procesanalyse dient juist als benchmarking, controle op eigen situatie ten opzichte van gelijken, historie of doelen. Beide kunnen op strategisch niveau zeer waardevolle inzichten bieden aan bestuurders om bedrijven winstgevend te houden.

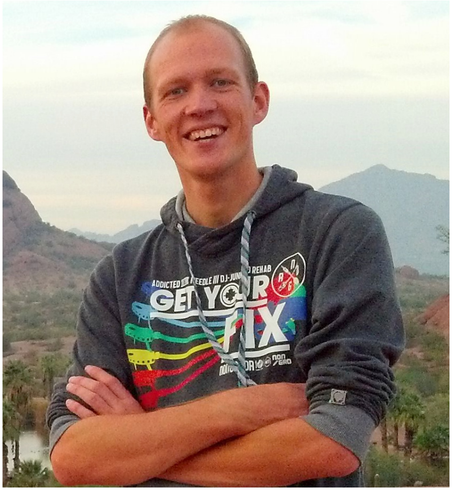
Oké gaaf! Maar welke informatie uit het werkproces is dan interessant om te visualiseren om van te leren of om op te sturen? Laten we de Universiteit als voorbeeld nemen. Zij ziet graag haar ambitieuze studenten zo spoedig mogelijk in hun studie doorstromen, maar wel zo, dat ze voldoen aan de inspectienormen, opdat de onderwijskwaliteit op peil blijft en tevredenheid onder de studenten blijft heersen. We nemen deze versimpeling van drie pijlers even als uitgangspunt: doorstroming, inspectie van onderwijskwaliteit en studenttevredenheid. Neem in gedachten wie er belang hebben bij inzicht in deze metingen. Zelf dacht ik aan: directie, docenten, maar ook studenten. Voor ieder van de doelgroepen kan je formuleren waarvoor deze informatie zou worden gebruikt. Het liefst zien we deze aangevuld in de vorm van 'use cases'. Hoe eenvoudig het misschien ook klinkt, het stellen van de vraag aan een gebruiker wat zijn 'use case' is voor de gestelde informatiebehoefte levert soms geweldige inzichten. Een gebruiker komt erachter dat de informatie eigenlijk overbodig is, of verkeerd gevormd (verkeerde visualisaties), niet aanwezig (niet geregistreerd), onbetrouwbaar (de gebruiker weet niet hoe de informatie tot stand is gekomen), of nog niet beschikbaar (niet vastgelegd of gemeten).

Bij het bouwen van je systeem zal je je ook mogen afvragen voor wie de informatie bedoeld is. Kan dezelfde informatie voor alle doelgroepen op eenzelfde wijze worden gevisualiseerd, of behoeven de verschillende groepen verschillende interpretatie, focus of misschien zelfs afscherming van gegevens. Daarbij mag je met al je ambitieuze ideeën wel nog rekening houden met budget, ontwikkelcapaciteit en technologische uitdagingen. Dus er zullen keuzes gemaakt moeten worden omtrent welke 'use cases' je gaat uitwerken.

Voor het gemak in deze casus betaalt de Universiteit en is zij op zoek naar sturingsinformatie voor hun dagelijkse directie. De doelen voor het komende jaar moeten worden opgesteld en de directieleden zijn voornamelijk benieuwd naar de uitdagingsgebieden voor het aanstaande jaar. Zij willen meten hoe dit gedurende het jaar (jaren) zich zal verbeteren op basis van metrieken zoals studentendoorstroming. Nu loop je bij het definiëren van doorstroming echter al snel aan tegen verschillende interpretaties. Wat bedoelen we met doorstroming, wat meten we dan eigenlijk? Je zou bijvoorbeeld de doorstromingstijd kunnen meten (van instroom tot uitstroom). Instroom meten we dan op basis van inschrijfdatum van de student, de uitstroom op uitschrijvingsdatum van een student. Nu zijn er echter studenten die een tussenjaar nemen en zich niet inschrijven. Hier eindigt mijn perfecte statistiek al. Maar er speelt nog veel meer. De doorstromingstijd geeft bijvoorbeeld ook niet aan in hoeverre er gewisseld is per studie.

Daarnaast weten we ook niet wat een doorstromingstijd van 6 jaar zegt wanneer we als gebruiker niet beschikken over de domeinspecifieke kennis. Wat is 6 jaar als er vaak ook andere studies worden gevolgd? Een langere doorstroming kan namelijk tal van oorzaken hebben die juist in het kwalitatieve vlak gemeten behoren te worden. Zo kan een student een tussenjaar nemen, werken naast zijn studie en nevenwerkzaamheden vervullen zoals een bestuursjaar. Het mag de student dan wel ten goede komen, het zal wel mede de bestuursmetrieken doen bepalen. Als hier rekening mee wordt gehouden, hoort de organisatie zich hiervan wel bewust te zijn voordat zij hierop gaat sturen. Informatie dient zich vaak aan als een simpel gegeven, maar wanneer je een analytische blik neemt op het gehele proces en de markt waarin het plaatsvindt dan mag je ook kritisch kijken wat we daadwerkelijk willen ontwikkelen.

Naast het definiëren van de metriek speelt ook de communicatie van de metriek een rol in effectiviteit. Zo schreef Darrell Huff, goeroe in statistische representatie, ooit over het intentioneel gebruik van informatievisualisatie ten behoeve van misinformatie, vaak bedoeld ter bevoordeling van het eigen product. Tegenwoordig zien we juist meer de deïntentionele verspreiding van misinformatie die plaatsvindt doordat de gebruiker of softwareleverancier met hun visualisatie juist niet communiceert wat zij eigenlijk bedoelde. Wanneer we dus visualiseren moeten we achterhalen wat er dient te worden gecommuniceerd met de grafiek, ofwel met welke reden we zo'n grafiek bekijken. Zijn we op zoek om studies onderling te vergelijken, knelpunten te ontdekken of aandachtsgebieden te vinden om op te focussen? Een tal van mogelijkheden die allemaal bepalend kunnen zijn voor het ontwerp van je visualisatie. We kunnen studies naast elkaar plaatsen om te zien welke studies het goed doen, of juist de probleemgevallen tonen. Om nog maar even te zwijgen over de beperkingen die het dashboard ons meegeeft. De Universiteit kent ruim 120 studies, hoe past dat op een dashboard zonder dat we de gebruiker overladen met informatie, een van de meest voorkomende fouten uit de jaren 90 volgens Stephen Few, die het fenomeen als "Dashboard Confusion" paradeerde. Het lezen van een goed doordachte, welgevormde grafiek is niet zo moeilijk, maar daar gaan dus wel veel beslissingen ten behoeve van effectieve communicatie aan vooraf. Het gebruik van een aantal vuistregels is dan ook niet onverdienstelijk. Grice's conversational maxims is daarvoor een mooi uitgangspunt. De theorie is overigens heel soft en filosofisch, maar het gedachtengoed kan je helpen als leidraad bij het evalueren van je grafiekontwerp. De theorie deelt zich in vier delen: kwantiteit, kwaliteit, relevantie en wijze.



Kwantiteit
- Maak jouw bijdrage zo informatief als noodzakelijk
- Maak jouw bijdrage niet informatiever dan noodzakelijk

Kwaliteit
- Vertel nooit dat waarvan je denkt dat het mogelijk onwaar is
- Vertel nooit dat waarvoor je onvoldoende bewijs hebt

Relevantie
- Vertel alleen de zaken die betrekking hebben op je gespreksonderwerp

Wijze
- Vermijd de obscuriteit van meningen
- Vermijd ambiguïteit
- Houd het kort
- Zorg voor orde en structuur

De fundamentele uitdagingen liggen zoals eerder genoemd in het vaststellen van je metrieken en correctief gebruik van visualisatie ten behoeve van correcte effectieve communicatie. Echter door de vele ontwikkelingen die nog in alle toepassingsdomeinen plaatsvinden, waardoor nieuwe gegevens beschikbaar komen in combinatie met de kennis die we over effectief visualiseren opdoen, besef je pas hoe experimenteel we eigenlijk nog bezig zijn bij de ontwikkeling van informatievisualisatie •

AUTHOR: **BART MARINISSEN**

Bitcoin

Bitcoin is garnering a lot of attention. Just recently, bitcoin earned its creator a Nobel Prize nomination. In tech circles, bitcoin has been making waves for a long time. It earned notoriety by enabling the black market of silk-road. Before that, it drew a lot of attention for its large value increases. All the while, those looking into the details of the implementation have marvelled at the ingenious techniques it uses.This has given rise to many other applications based on the same techniques.

The genius of bitcoin lies in decentralization. Bitcoin eliminates the single point of control and trust that other currencies depend on. On top of that, it offers some anonymity, though this anonymity has some limits. Many people have opinions on whether this is a good or a bad thing. However, we will only look at how bitcoin works. We start with a very brief overview. Bitcoin is based on a network of *nodes* that all store the list of all transactions that ever happened (called the block chain). To transfer bitcoin, you create a transaction that must be cryptographically signed, and offer this to the nodes. These nodes then compete to process your transaction, because upon doing so, they collect a reward.

We start off with a very simple question, what is a bitcoin? Bitcoins are not actual physical coins; you don't 'own' an actual bitcoin. They also differ from a bank account. There is no central database storing how much money you have a right to. Instead, to own Ba bitcoin means to have proof someone sent you bitcoin which you yourself haven't spent yet.

Here, it is your responsibility to prove bitcoin was sent to you. It is the system's responsibility to ensure you haven't spent this bitcoin already. The *proof* you have to provide is *a cryptographic signature* of the transaction.

Transactions

Currencies are made to be transferred. For physical currency, transactions are easy. However, for bitcoin they are more complicated. In fact, the entire point of bitcoin is keeping track of all transactions that happened. As we saw before, you don't *own* a bitcoin, you have proof a transaction was intended for you. So how do transactions work in bitcoin?

A bitcoin transaction consists of two lists: the outputs and the inputs. An output gives an amount of bitcoin to someone, as such, it contains the amount to transfer, and information identifying the recipient. An input is a reference to the output of a previous transaction with proof it was intended for you. A transaction is valid if it meets three requirements:

1. All inputs must reference an output, which has not yet been spent (also called a UTXO, unspent transaction output).
2. All inputs must contain valid proofs. That is, the proof has to be checked against the identifying information of the referenced output.
3. The sum of all outputs must not exceed the sum of all inputs.

This concept of UTXO is crucial to understanding bitcoin. Spending bitcoin means 'destroying' certain UTXO and creating certain new UTXO. Here, destroying UTXO means changing the status of an output from unspent to spend. As such, it is really important to keep track of all the UTXO currently in circulation. It is the only way to check condition 1.

Comparatively, condition 2 and 3 are much easier to check. Condition 3 is simple arithmetic and condition 2 only requires access to a few specified transactio ns. The main question of this section is how exactly condition 2 works. That is, what the *identifying information* in an output is and how to check this against the *proof* of an input.

Before we get to that there are some peculiarities that deserve mentioning. Firstly, you cannot spend part of a UTXO. If you want to spend less, you have to give yourself some change. Secondly, requirement 3 does not require equality, you could have some 'left over' input. Why would you ever have left over input, and where does it go? The leftovers form a *fee* and are given to whomever processes the transaction. You can offer a fee as an incentive to process a transaction earlier. Finally, every transaction recursively needs a valid previous transaction. This recursion ends at the *coinbase*. This is an artificial transaction that does not require any inputs as such, it creates new bitcoin from nothing. Therefore there are heavy restrictions on the creation of these coinbase transactions. We will see later what these restrictions are when we look at the block chain.

Now, for the main question: how do we meet condition 2? That is, what should the relation be between the identifying information in an output, and the proof in an input that spends it? In general, the identifying information takes the form of a *challenge*. The proof of an input should then correctly *answer* this challenge. There are many possible ways to construct such a challenge, however the vast majority of transactions use a variant called *pay to pubkey hash*. This is an enhanced version of the older *pay to pubkey*.

We will start with *pay to pubkey*. This protocol is build around cryptographic signatures. In turn, these are build around hashes and public-key cryptography. A signature is made with a public-private key pair, and *signs* some data. Cryptographic signatures have a crucial property: *signing* data can only by done with the private key, whereas *checking* a signature only requires the public key. Moreover, a signature for one *file* cannot be used to create a signature for another file.

Pay to pubkey then works as follows: the identifying information contains the recipient's public key, and instructions to use pay to pubkey. The answer then entails a signature of the transaction containing the referenced output. Only the person who has the private key that corresponds to the public key in a UTXO can create this signature. However, anyone can verify the signature and thus the transaction.

Pay to pubkey hash is a modern variant that uses the same signature concept. However, the identifying

Cryptographic hash

A cryptographic hash is a function *H* that takes input of any length, called a *message* and returns a fixed size output called a *digest*. Furthermore, it needs to fulfil the following properties:

- Computing it is easy
- It is practically impossible to use the digest to retrieve the message
- It is practically impossible to find two different messages with the same hash. (also known as finding a *hash collision*)

The main use of a hash is to provide a check. If you know to expect a message with digest *D* and get a message *M* you can simply check of $H(M) = D$. This is an easy operation, and you can be sure this is the intended message. One very common property of a hash is that a small change in the message causes a large change in the digest. This has the added effect of hashes being semi-random.

information no longer contains the public key used to verify the signature. Instead, only the hash of a public key is given this is called an *address*. It is then up to whomever creates the input to also provide the public key. When checking such a transaction, you don't just check the signature, you also check whether the public key matches the given hash. The reason for this change is simple: an address is much shorter than a public key. This makes it easier to instruct someone to send you bitcoin.

It should be noted that it is best practice to use each address only once. This means you have to keep track of all your addresses and their corresponding public-private key pairs. This is done with a *wallet*. By using each address only once, you gain some anonymity; all transactions intended for you have different addresses. However, it is still a good bet that the inputs of a transaction belong to the same person.

There are many other transaction types. One can for example require that two out of three people provide a signature, enabling escrow by a third party. This is why the identifying information is accompanied by the type of challenge used. In fact, this is all based on a limited (non-Turing complete) scripting language.

Block chain

Now that we've seen how transactions work, we get to the fun part. How can we check if an input references a UTXO. This is actually an old problem with an old solution: a ledger. A striking parallel is keeping track of who has title to what piece of land. Transactions can split up and combine pieces of land and you can only give a piece of land if you ever received it and haven't since given it away.

However, traditional ledgers have some limitations stemming from their centralized nature. Now, a ledger needs to be centralized to ensure consistency, which is a hard requirement. However, centralization creates a single point of failure and thus, a single point of trust. Block chain aims to remove this single point of trust by creating a distributed ledger. Notably, block chain really implements a ledger. It can thus be used to keep track of who has title to something. This idea is implemented in NameCoin.

This brings us to the field of distributed consensus. The general idea here is as follows. A network of *full nodes* all keep the ledger. From this, they can derive the current UTXO pool. To transfer Bitcoin, you create a transaction and offer it to one or more of these nodes. They then check the transaction against their UTXO pool. If it is valid, they forward the unconfirmed transaction to the other nodes. Roughly every 10 minutes, a new *block* of transactions is *mined* by someone and sent to the full nodes. The full nodes check this block and ideally, if it is valid, enter its transactions into the ledger and update their UTXO pool.

Besides a list of transactions, a block also contains a header. Crucially, this header references the previous block on which it is based. This way, the blocks form a chain. Or more accurately, every block has a chain of blocks on which it depends. All blocks actually form a tree. Two blocks could have the same parent. At the root of this tree lies the *genesis block*. This block does not reference a *previous* transaction. Every correct node must consider the genesis block valid.

However, if blocks form a tree, we now have multiple ledgers. In fact, every leaf corresponds to a different *valid* ledger. To solve this, the rule for a full-node is as follows: always believe the longest chain. So, if one receives a valid block that references another block than the end of the current longest chain, we add it to the tree of blocks. Unless this now becomes the end of the longest chain, nothing further happens. However, when this new block does suddenly become the longest chain, history gets rewritten.

That a ledger allows rewriting history is a big deal. It seems like our solution to the single point of trust is to simply trust that no-one will ever want to rewrite history. This is blatantly not the case; rewriting history is the basis of the only practical attack (that is, an attack that does not require breaking cryptography) on bitcoin: the double-spend. It goes as follows: spend some UTXO and get the goods in return, then rewrite history to erase the original transaction, and create a new transaction spending the same UTXO. It is crucial to create the new transaction, otherwise the old transaction can simply be entered again by anyone. After all, it is a valid transaction referencing a UTXO.

However, rewriting history is only as easy as creating enough valid blocks is. No one said creating a valid block was easy. Actually, it is really, really hard to create a valid block. A valid block requires *proof of work*. To see what proof of work means, let's look at blocks and block validation more closely. A block contains a header and a list of transactions. The header has the following fields:

version	version of the protocol used
timestamp	time block was created
previous block	hash of the previous block
transactions hash	hash of all transactions of the block (actually the root of the merkle tree of the block)
target	maximum value for hash of this block
nonce	free value to make meeting the target possible

Public-key cryptography

Public-key cryptography uses a pair of keys as opposed to normal cryptography's one key. This is called a public-private key pair (k, k'). Generally, one of these is called the *public key* and the other the *private key*. The encryption and decryption algorithms are the same. We shall write $C_k(M)$ to denote en- and decrypting with key k. Unlike normal encryption, you do not decrypt with the same key as a message was encrypted. Instead, the public and private keys are each other's decryption keys. So $C_{k,\text{pub}}(C_{k,\text{priv}}(M)) = M$. To ensure security, it is practically impossible to retrieve M from $C_k(M)$ without the corresponding public or private key.

One interesting application is forming **cryptographic signatures**. A cryptographic signature uses a public-private key pair, and *signs* some data. The procedure is as follows: Given message M, hash it, and encrypt that with your private key. So the signature is given by: $S = C_{k,\text{priv}}(H(M))$. One can check a signature S for message M and public key k_pub by checking that $C_{k,\text{pub}}(S) = H(D)$. Only someone with the private key can compute S for data D yet anyone with access to the public key can verify the signature.

For a block to be valid, all of the containing transactions must be valid and the header must be valid. The main requirement for the header is that its hash is lower than its target. This is the proof of work. Furthermore, the timestamp of a block must also make sense. Finally, the target depends deterministically on the block chain. Those looking to create new blocks don't just get to set their own target. Before we get to proof of work, there is another very important observation: the reference to the previous block is stored as a hash. This means that changing any block invalidates any block further down the chain. Changing history really requires building an entire new chain.

The key to proof of work is that hashes are practically random. The hash used here is two rounds of sha-256, which has 256 bits of output. This means the chance of getting a hash lower than target T is $P = 2^{256} / T$. This, scaled by a factor 2^{32} is called the difficulty: $D =$

$2^{32} P = 2^{224} / T$. This means that, on average, one needs to try $1/P = T / 2^{256}$ hashes before meeting the target. As such, the target, combined with the total hash rate of all nodes, determines how long it takes to *mine* a block. As said, the aim is for this to take 10 minutes. As such, the difficulty is recomputed once every 2016 blocks that is, roughly every 4 weeks. When recomputing the target, one simply looks at the timestamps to determine how long it took to mine the last 2016 blocks. If we call this t_b, the new target is given by: new target = target * 4 weeks / t_b. At the time of writing, the difficulty is 72722780643. This corresponds to a hashing rate of $(72722780643 * 2^{32})$ / 10 minutes ≈ $5.2 * 10^{17}$ hashes per second.

Interestingly, the current implementation has an off-by-one error, causing only the last 2015 blocks to be taken into account when calculating the target. Fixing this is very hard, not because fixing the code is hard, but because consensus needs to be maintained. Nodes that disagree on the difficulty will never accept the same block. This illustrates a core fact of bitcoin. One should follow consensus rather than the specification. This makes backwards incompatible changes very hard to implement. It is also said that a change requires a *hard fork*, as going forward, there would be two chains. One of those who made the change and one of those that didn't.

Crucially, proof of work does not make rewriting history impossible. It simply makes it more difficult. Luckily, the deeper a block lies in the chain, the harder it is to replace. Each block added to the chain in some sense *confirms* all the transactions in that chain. The defence against a double spend is then simple. Wait until your transaction has been confirmed often enough before handing over the goods. How often is enough is determined by what risk you find acceptable. A general guideline is to wait for 6 confirmations. This gives someone with 10% of the computing power a less than 0.1% chance of rewriting the last 6 blocks. [1]

Mining

We have seen how to *mine* for new blocks, and we saw that a lot of effort is going into this, but we didn't see why. Unless it is to execute a double spend, why would anyone ever mine? There are actually two reasons, and we have mentioned both. Firstly, there are the fees. Remember that any left overs in a transaction are a fee for processing a transaction; this fee goes to the miner. Secondly, and for the moment more importantly, there is the base reward. Recall that we mentioned that bitcoins get created through a *coinbase* transaction. Every new block contains a single coinbase transaction. Whoever mines that block gets to decide the outputs of the coinbase. However, the total value is fixed at a base amount + all the fees. Notably, the UTXO of a coinbase only becomes spendable after it is 100 blocks deep into the chain.

The base amount of a coinbase started at ฿50 and halves every 210,000 blocks (roughly every 4 years). As such, all bitcoin that will ever be created is a geometric series converging to ฿21,000,000. This is very much an intentional feature of bitcoin. It was modelled after the rate at which commodities like gold are found. This is where the term *miner* comes from. As the base reward for mining decreases, so does the incentive it offers to mine. Eventually, the main motivation for mining will be to gather the fees. Much like transitioning from a production based economy to a service based economy.

Unless you own a significant portion of the total hashing power, mining on your own is very inconsistent. You only get a pay out when you actually manage to mine a block. To counteract this, miners band together in *mining pools*. They work together, and share their pay outs proportionally to how much each of them contributes.

However, miners in a pool are not blindly trusted to report their own hash rate. Instead, each pool has a *pool master*. Everyone is working on a block that pays out to this pool master. However, they are not mining to meet the current global target. Instead, they are mining to meet a *pool target* usually 2^{224} - 1. Each time they mine a block that meets the pool target, they submit it to the pool master. If the block also happened to meet the global target, the pool master can submit it, and pays out proportionally to how many blocks a miner submitted. Not coincidentally, on average one needs to try $T / 2^{224} = D$ hashes to find one that meets the pool target, exactly the difficulty. You might wonder why you would ever submit a block that meets the global target to the pool master. You could submit it yourself! Luckily (or sadly) there is no point, the coinbase of the block you mined still pays out to the pool master.

One issue with mining pools is that they centralize. Pool miners do not need to check anything in the block. In fact, they do not even need the whole block, just its header (and a small 'summary' of the transactions). This means there are fewer checks on the data. The danger is not just imaginary; there were real concerns in 2014 when one pool controlled 40% of the hashing power. For reference, this gives a double-spend attack on a 6-deep transaction a roughly 50% chance of succeeding. [1]

State of bitcoin

In general centralization forms a problem for bitcoin. There is the ever looming threat of a mining pool controlling too much of the hashing power. Another threat is the ever decreasing number of full-nodes. Full-nodes are crucial in keeping track of the block chain and getting transactions to miners. However, there is no incentive to run a full node. The only reason to run one is either philanthropy or to make sure you can always reach a full node. As such, the amount of full nodes has steadily declined from 9,500 in March 2014 to 5,100 at the time of writing.

Another issue is the maximum amount of transactions per second that bitcoin can process. We can calculate this based on the rate of new blocks being created (1 per 10 minutes), the maximum block size (1MiB) and a minimum transaction size (166B). This gives a maximum rate of 10 transactions per second. Compare this to the current rate of 115 per second for PayPal and 2000 per second for VISA. Changing this would require a hard fork, as currently blocks over 1MiB are considered invalid. However, there is debate on whether larger blocks are even desirable.

The proof of work concept has also received quite a bit of criticism. Initially, most miners used their CPU to calculate the hashes. It did not take long until people realized that GPUs (graphics cards) were orders of magnitude faster at calculating hashes. More recently, people found that calculating the sha-256 hash is so specific a task, you can create specialized electronics to do it, these are called ASICs for Application Specific Integrated circuits. These days, running anything other than an ASIC mining rig is simply not profitable. Sadly, this means mining is no longer something anyone can start and stop doing. Instead, it is something you buy into.

Moreover, proof of work requires a huge amount of computational and electrical power. As such, many alternative cryprocoins (i.e. altcoins) have been suggested. Litecoin and Ethereum use proof of work with a different hashing algorithm. The idea here is to foil ASICs by making the algorithm memory hard. Computation times are then bound by the memory speed. In the case of litecoin, ASICs have already come to dominate mining. Ethereum is still new, but where Litecoins algorithm script uses 16MiB of memory, Ethereums algorithm Ethash uses 1GiB of memory, so there is hope yet. Other cryptocoins have moved away from proof of work entirely. Some move to a *proof of stake* system. Here, each coin you own gives you a chance to mine a new block. The idea being that having coins means you have a stake in the system remaining credible. Finally, there is the option of moving away from making blocks hard and instead make the creation of illegitimate blocks impossible. Such systems would use techniques derived from the byzantine general's problem.

However, despite all of these alternatives being created, bitcoin is still going strong as the largest cryptocoin. Some even say bitcoin is the only cryptocoin to be taken seriously. Certainly it has the largest user base by far, making it the most trustworthy. Only time will tell whether the community at large will be swayed by an altcoins improvement, or whether they will stick with the proven bitcoin •

References

[1] https://bitcoil.co.il/Doublespend.pdf

AUTHOR: **MARTINE SCHROOR**

Surf and Turf

As I am currently living in Australia, I decided to tell about a typical Aussie dish called 'Surf and Turf'. This recipe always highlights two sources of proteins, meat and seafood (usually crustaceans), and varying side dishes such as potatoes in this one. Not everybody likes the idea of the combination of seafood and meat, but you should give it a try.

Ingredients

- 2 tbs butter
- 2 cloves garlic crushed
- 2 tbs plain flour
- 1 pinch paprika
- Salt to taste
- Pepper to taste
- 600 ml thickened cream warmed
- 500 g green prawns patted dry with paper towel peeled
- 1/4 cup (5-10 g) fresh curly parsley chopped to taste
- 4 rump steak (beef)
- Potatoes

Clean the potatoes and bring them to the boil, cook these for approximately 20 minutes. Meanwhile put the frying pan on a high heat and sear the steak on both sides until cooked to the preferred consistency. Melt butter on low heat in a medium sized saucepan. Add garlic and cook for 1 minute, stirring. Add flour, paprika, salt and pepper and stir to combine. Increase heat to medium high. Gradually add 500 ml of the warm cream in batches, briskly stir to combine before adding more cream. Add prawns and parsley, and simmer until prawns are just cooked through. Add up to 100 ml of the reserved cream if too thick. Serve over cooked steak with potatoes •

http://www.bestrecipes.com.au/recipe/surf-and-turf-with-garlic-cream-sauce-L12379.html {kxa} De dataspecialisten

Software Innovations

KxA software innovations is gevestigd in de provincie Groningen. Het is een uniek bedrijf dat innovatieve, gekke, grote, kleine, duurzame, sociale, maar natuurlijk ook normale maatwerk software-opdrachten uitvoert. De overeenkomst tussen al deze projecten is dat het gaat om data in alle soorten en maten, bijvoorbeeld:

Het Nederlandse verkeer in 400 miljard metingen toegankelijk opslaan

In een stal het gedrag van koeien monitoren

Software ontwikkelen voor de gigantische SKA radiotelescoop

van BIG BANG tot

Hulpverleners in de zorg ondersteunen met VCA

Werken bij KxA

Bij ons vind je allerlei achtergronden (natuurkunde, informatica, AI, etc). Iedereen deelt het enthousiasme voor softwaretechniek en wat je daar allemaal mee kunt doen.

We hebben regelmatig afstudeeropdrachten, stageplekken én vacatures. Je krijgt hierbij een opleidingstraject om je helemaal in ons vakgebied te bekwamen. Het vervoer naar Visvliet kan geregeld worden.

Ben jij geïnteresseerd in het werken bij een High Tech bedrijf? Kijk dan eens op **www.kxa.nl**, of neem contact met ons op via mulder@kxa.nl

AUTHOR: **THE EDITORS**

Previous Brainwork

Een kruisgetalpuzzel

Many of you spend a lot of time on the previous brainwork. The lucky winner of this brainwork is Corina van der Lei! Congratulations for finding the correct solution and have fun with your “Wiskundeboek”. For those who want to check their own answer, the solution is given below.

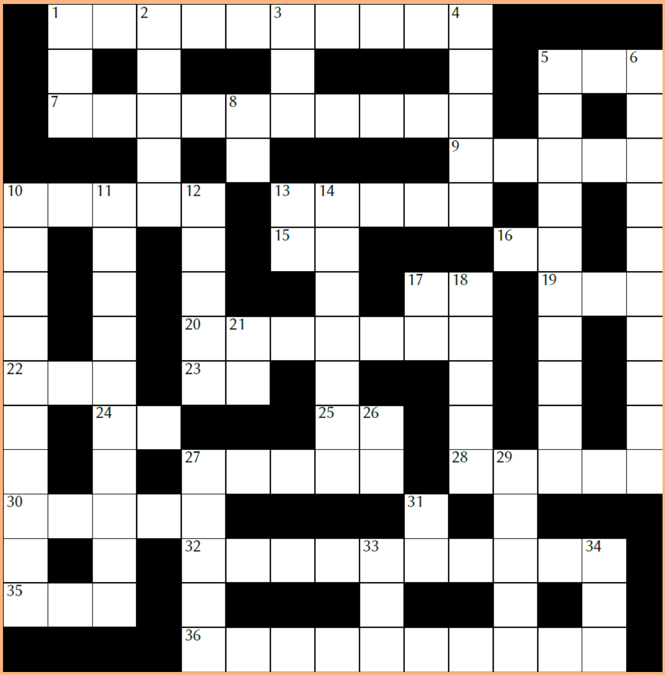
The solution

^{1}1	^{2}2	^{3}1	■	^{4}1	^{5}3	^{6}3	^{7}1	■
^{8}5	0	9	■	^{9}7	0	6	3	^{10}8
^{11}4	2	4	^{12}1	■	■	^{13}1	4	4
■	■	^{14}2	9	^{15}7	^{16}1	■	^{17}2	1
■	^{18}5	5	5	5	5	^{19}5	5	■
^{20}3	6	■	^{21}1	0	9	8	■	■
^{22}1	4	^{23}5	■	■	^{24}7	7	^{25}7	^{26}6
^{27}6	2	0	^{28}0	^{29}3	■	^{30}1	2	7
■	^{31}1	0	2	4	■	^{32}6	6	6

New Brainwork

A cross number puzzle

Do you also have that problem, that you have completed the Christmas puzzles even before Christmas has even started. Well, if you do, we may have something that might keep you busy a while longer: another cross number puzzle. As editors, we already had lots of fun with this problem and we hope that this one helps you through the holidays. About the notation: 4D, for example, means the fourth number going down and 24A means the twentieth fourth across. The solutions can be handed in before the 29th of January 2016 at perio@fmf.nl. Among the right solutions we will select the winner of a The Colossal Book of Mathematics by lottery.



The challenge

ACROSS

1. 4D multiplied by 18D. (10)
5. A multiple of 101. (3)
7. The difference between 10D and 11D. (10)
9. A palindromic number containing at least one 0. (5)
10. Subtract 24A multiplied by 24A backwards from 100000. (5)
13. Subtract 8D from 35A then multiply by 17A. (5)
15. Multiply this by 13D to get a perfect number. (2)
16. The product of two primes. (2)
17. A triangular number. (2)
19. A factor of 6D. (3)
20. 30A more than the largest number that cannot be written as the sum of distinct fourth powers. (7)
22. The sum of seven consecutive primes. (3)
23. When written in Roman numerals, this number is an anagram of XILXX. (2)
24. The largest prime factor of 733626510400. (2)
25. A square number. (2)
27. The product of all the digits of 7A. (5)
28. A multiple of 107. (5)
30. Unix time at 01:29:41 (am) on 2 January 1970. (5)
32. When written in a base other than 10, this number is 5331005655. (10)
35. The smallest number which is one more than triple its reverse. (3)
36. All but one of the digits of this number are the same. (10)

DOWN

1. 700 less than 3D. (3)
2. The sum of this number's digits is equal to 16. (5)
3. A Fibonacci number. (3)
4. This is the same as another number in the cross number. (5)
5. A square number containing every digit from 0 to 9 exactly once. (10)
6. This number's first digit tells you how many 0s are in this number, the second digit how many 1s, the third digit how many 2s, and so on. (10)
8. The lowest prime larger than 25A. (2)
10. The largest prime number with 10 digits. (10)
11. A multiple of 396533. (10)
12. If you write a 1 at the end of this number then it is three times larger than if you write a 1 at the start. (5)
13. Multiply this by 15A to get a perfect number. (2)
14. The factorial of 17D divided by the factorial of 16A. (7)
17. The answer to the ultimate question of life, the universe, and everything. (2)
18. A multiple of 5. (5)
21. The number of the D clue that has the answer 91199. (2)
26. The total number of vertices in all the Platonic Solids (in 3D). (2)
27. Two more than 29D. (5)
29. The first and last digits of this number are equal. (5)
31. A multiple of 24A. (2)
33. Each digit of this number is a different non-zero square number. (3)
34. A square number. (3)